
VMS Version 5.5-2 Release Notes

Order Number: AV-PQZAB-TE

August 1992

This document describes software problems, corrections, restrictions, documentation changes, and new hardware support that pertain to Version 5.5-2 of the VMS operating system.

Revision/Update Information: This is a new manual.
Software Version: VMS Version 5.5-2

**Digital Equipment Corporation
Maynard, Massachusetts**

August 1992

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

© Digital Equipment Corporation 1992.

All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: ALL-IN-1, CI, DECdtm, DECmcc, DECnet, DEC PHIGS, DECwindows, DEQNA, Digital, HSC, KLESI, PATHWORKS, Q-bus, Q22-bus, RA, RSX, RRD42, TMSCP, TURBOchannel, UNIBUS, VAX, VAX Ada, VAX C, VAX DOCUMENT, VAXstation, VMS, and the DIGITAL logo.

BASIC (Dartmouth Structured Basic) is a registered trademark of the Trustees of Dartmouth College, D.B.A. Dartmouth College.

UNIX is a registered trademark of UNIX System Laboratories, Inc.

ZK5961

This document was prepared using VAX DOCUMENT, Version 2.1.

Contents

Preface	xi
1 General User Release Notes	
1.1 Digital Command Language (DCL)	1-1
1.1.1 EDIT/TPU Command /NOWORK Qualifier—Problem Corrected	1-1
1.1.2 Error Message Displayed When Non-Modem Port Is Set to Modem . . .	1-1
1.1.3 ON Condition Response—Problem Corrected	1-1
1.1.4 READ Command Error Message	1-2
1.1.5 SET DEFAULT and SHOW DEFAULT Commands—Behavior Changed	1-2
1.1.6 SET MESSAGE Command Changed—Problem Corrected	1-2
1.1.7 SET TERMINAL/SPEED Command—Problem Corrected	1-3
1.1.8 SET TERMINAL/COMMSYNC Command—Support Added	1-3
1.1.9 SHOW DEVICE/FULL Command—Problem Corrected	1-3
1.1.10 TYPE Command /PAGE Qualifier—Enhanced Screen Usage	1-4
1.2 Mail Utility (MAIL)	1-4
1.2.1 MAIL Sends and Receives Compound DDIF and DTIF Documents . . .	1-4
1.2.2 Missing Error Message in MAIL—Problem Corrected	1-4
1.3 VAXstation 4000 Keyboard Problem	1-4
1.4 VAX Text Processing Utility (VAXTPU)	1-4
1.4.1 EVE Key Name Definitions—Problem Corrected	1-4
1.4.2 FILL Built-In Procedure—Problems Corrected	1-5
1.4.3 Tabs in Overstrike Mode—Problem Corrected	1-5
1.4.4 Work Files—Problem Corrected	1-5
2 System Manager Release Notes	
2.1 Allocation Classes in VAXcluster Configurations—Restriction	2-1
2.2 Audit Analysis Utility /SELECT=STATUS=FAILURE Qualifier—Problem Corrected	2-1
2.3 AUTOGEN Command Procedure Notes	2-1
2.3.1 AUTOGEN Problem When Executed from SYSMAN	2-1
2.3.2 Specifying a Minimum Required Age for AUTOGEN Feedback Data	2-2
2.3.3 Specifying the Number of Ethernet Adapters for AUTOGEN	2-2
2.3.4 Specifying the Number of VAXcluster Nodes for AUTOGEN	2-2
2.3.5 Suggested Workaround to SYSMAN Value Error	2-2
2.3.6 TMSCP_LOAD Parameter—Problem Corrected	2-3
2.4 Backup Utility (BACKUP)	2-4
2.4.1 BACKUP Denies Privileged Users Access to a Tape	2-4
2.4.2 BACKUP/LIST Operations on a Save Set	2-4
2.4.3 Data Compactions on Tape Drives	2-4
2.4.4 Extending Index Files	2-4

2.4.5	Message Status Codes Restored	2-5
2.4.6	Multivolume Logical Names	2-5
2.4.7	Multivolume Save Operations from a Subprocess	2-5
2.4.8	Saving and Restoring Alias Directories	2-5
2.4.9	Tape Volume Protection	2-6
2.5	Batch and Print Queuing System	2-6
2.5.1	ALL-IN-1 and Batch and Print Queuing System—Interaction Problems Corrected	2-7
2.5.2	Chance of Queuing System Shutting Off During Node Shutdown	2-7
2.5.3	Corruption Detection for Queues	2-7
2.5.4	DEFINE/CHARACTERISTIC Command Allows Only One Name to a Number	2-8
2.5.5	VAX Distributed Queuing Service (DQS) Print Symbiont—Suggested Workaround	2-8
2.5.6	Generic Queues with Implicit Target Lists—Change in Behavior	2-9
2.5.7	Queue Manager CPU Limit—Problem Corrected	2-10
2.5.8	Queue Manager Stop_Pending Status—Problem Corrected	2-10
2.5.9	Queue Manager SYMDEL Message—Problem Corrected	2-10
2.5.10	Saving Information in the Queue Database—Problem Corrected	2-10
2.5.11	SUBMIT/USER and PRINT/USER Commands Provide Incorrect Account Name—Problem Corrected	2-11
2.5.12	SYNCHRONIZE Command—Problem Corrected	2-11
2.6	CLUSTER_CONFIG.COM Command Procedure—Updated to Read Undefined System Disks	2-11
2.7	Data Corruption on Disk with New Allocation Class	2-11
2.8	DECnet-VAX Notes	2-12
2.8.1	Adjusting NETACP Quotas with NETACP\$ Logical Names	2-12
2.8.2	Cluster Synchronization Improved	2-13
2.8.3	Event Logger—Problem Corrected	2-13
2.8.4	EXECUTOR NODE Corruption—Problem Corrected	2-13
2.8.5	FDDI and Token Ring Device Support—Problem Corrected and Improvements Added	2-13
2.8.6	FDDI Line Parameters for Phase IV Network Management	2-14
2.8.6.1	SHOW LINE Display Modified	2-16
2.8.7	NCP Command Prevents Unauthorized Connections to the Mail Server	2-17
2.8.8	NCP Commands for DNS Interface	2-17
2.8.9	NETACP Allocation of Node Counter Blocks Changed	2-18
2.8.10	NETDRIVER PATH SPLIT POLICY Call—Problem Corrected	2-19
2.8.11	NETACP Endnode Failover—Problem Corrected	2-19
2.8.12	NDDRIVER Queue Corruption—Problem Corrected	2-19
2.8.13	NCP Module Configurator—Problem Corrected and Improvements Added	2-19
2.8.14	NCP SHOW KNOWN NODES Display—Problem Corrected	2-20
2.8.15	Patched Images for DNS Version 1.1	2-20
2.8.16	Phase IV Support for DSW-21 and DSW-41/42 Devices	2-20
2.9	DECwindows	2-20
2.9.1	Dashed Lines Drawn Incorrectly—Problem Corrected	2-20
2.9.2	DECwindows Monitor Density Is Now Definable	2-21
2.9.3	DECwindows Server Incompatibility with DEC PHIGS	2-22
2.9.4	DECwindows Use of SMP with DECsound and SODRIVER	2-22
2.9.5	SoftPC with DECwindows Server—Keyboard Problem	2-22
2.9.6	Tiled Operations Drawn in Wrong Window—Problem Corrected	2-23
2.10	Delta/XDelta Utility (DELTA/XDELTA)	2-23

2.10.1	Delta/XDelta Utility—Booting New VAX Computers	2-23
2.11	Disk Class Drivers	2-24
2.11.1	MSCPCLASS Error—Problem Corrected	2-24
2.11.2	Opcode for MSCP Message Improperly Set—Problem Corrected	2-24
2.11.3	Shutdown Error While Copy Operation in Progress—Problem Corrected	2-24
2.11.4	Volume Shadow Set Members Race Condition—Problem Corrected	2-24
2.12	Distributed Lock Manager	2-24
2.12.1	Block Transfer to an Unavailable Node Caused System Failure— Problem Corrected	2-24
2.12.2	Lock Range Value Corruption—Problem Corrected	2-24
2.12.3	Remastering Routine Race Condition—Problem Corrected	2-24
2.12.4	RSPFATAL Status Caused System Failure—Problem Corrected	2-25
2.12.5	System Parameter Added to Lock Manager	2-25
2.13	InfoServer Client—Startup Behavior Change	2-25
2.14	LAT Not Supported on DEQNA	2-25
2.15	Mount Utility (MOUNT)	2-25
2.15.1	Bound Volume Set Problem Using Volume Shadowing Phase I and Phase II—Suggested Workaround	2-25
2.15.2	MOUNT Commands Delayed Other Nodes—Problems Corrected	2-26
2.15.3	MOUNT Command Caused CPUs to Halt—Problem Corrected	2-26
2.15.4	Shadow Sets Improperly Allowed—Problem Corrected	2-26
2.15.5	Shadow Set Failure with Two-Member Volume Shadow Set—Problem Corrected	2-26
2.15.6	Shadow Set Logical Names Were Improperly Defined—Problem Corrected	2-27
2.15.7	Shadow Set Members Failed—Problem Corrected	2-27
2.15.8	Tape Compaction Problems for TA90E/91—Problem Corrected	2-27
2.15.9	Tape Density Improperly Set—Problem Corrected	2-27
2.15.10	TUDRIVER During Mount Operation—Problem Corrected	2-27
2.16	New Operating System License Types	2-27
2.17	Operator Communication Manager (OPCOM) Restart	2-28
2.18	PATHWORKS PC Clients Can Interfere with InfoServer Clients	2-29
2.19	POSIX Version 1.0 Users Should Upgrade to POSIX Version 1.1	2-29
2.20	RMS Journaling	2-29
2.20.1	Invalid Journals—Workaround	2-29
2.20.2	RMS Journaling with DCL Command RECOVER—Problem Corrected	2-29
2.21	SET PREFERRED PATH Problem—Suggested Workaround	2-30
2.22	System Management Utility (SYSMAN) Server Hang	2-30
2.23	System Parameter LGI_BRK_TERM—Problem Corrected	2-30
2.24	System Symbol Definitions for High-Level Languages	2-30
2.25	VAX 6000-600 Warm Start for Systems with Battery Backup—Problem Corrected	2-31
2.26	VAX 62xx and 63xx Systems with KLESI or UNIBUS Adapters Enabled	2-31
2.27	Verify Utility (VERIFY)	2-31
2.27.1	File Header Identifier Error—Problem Corrected	2-31
2.27.2	Lost File Header Repair Caused Corruption of Data—Problem Corrected	2-32
2.28	VAXstation 3100 Model 80 and VAXstation 4000 Model 60—Problem Corrected	2-32
2.29	VMS Upgrade and VMS Update Procedures—Checking Directories	2-32
2.30	VMS Volume Shadowing	2-33

2.30.1	Improved Performance for Merge and Copy Operations	2-33
2.30.2	Effects on Performance	2-34
2.30.3	Enabling Assisted Merge Operations	2-35
2.30.4	Enabling Assisted Copy Operations	2-37
2.30.5	Controlling the Shadowing Performance Assists on HSC Controllers	2-39
2.30.6	VMS Volume Shadowing Restrictions	2-40
2.30.6.1	Recommended Version for VAXclusters, Volume Shadowing, and Striping	2-40
2.30.6.2	Using VMS Volume Shadowing in a Mixed-Version Cluster	2-40
2.30.6.3	Memory Requirements for VMS Volume Shadowing	2-41
2.30.6.4	Unnecessary Merge During System Reboot—Suggested Workaround	2-42
2.30.6.5	Connection Loss During Shadowing State Change May Cause Bugcheck	2-42
2.30.7	Supported Shadow Sets Maximum Increased to 75	2-42
2.30.8	SHOW DEVICES Display Incorrect for Merge Copy	2-43
2.30.9	Assisted Copy Operation Resets Incorrectly After Minimerge	2-43
2.31	VMS Version 5.4 to Version 5.5 Queue Database Upgrade Problem Resolution	2-43
2.31.1	Problem Information	2-43
2.32	XQP and File System in Mixed-Version VAXclusters	2-45

3 Programmer Release Notes

3.1	DSA Multiple Controllers Timeout Errors—Problem Corrected	3-1
3.2	LATSYM Error During Hangup Event—Problem Corrected	3-1
3.3	Mail Utility (MAIL)—Callable Mail No Longer Loses System Privileges	3-1
3.4	Mailbox Driver EXE\$\$NDEVMSG Routine Error	3-1
3.5	Pseudoterminal Driver Control Connection Routines	3-2
3.6	Remote Port Command Procedure Sample	3-2
3.7	RMS Journaling	3-2
3.7.1	Access to a Remote Indexed Read-Only File Can Fail—Problem Corrected	3-2
3.7.2	Deadlock Among Multiple Detached Recovery Servers—Problem Corrected	3-3
3.7.3	RMS Journaling Detached Recovery—Problem Corrected	3-3
3.7.4	TID Format in Transaction Error Messages	3-3
3.8	Screen Management Run-Time Library (SMGRTL)	3-3
3.8.1	BLOCK_BORDER—Problem Corrected	3-3
3.8.2	Incorrect Output with Terminal Device Set to VT80—Problem Corrected	3-3
3.8.3	Input Routine—Problem Corrected	3-3
3.8.4	Line Wrapping Outside Virtual Display—Problem Corrected	3-4
3.8.5	SMG\$ Viewport—Problem Corrected	3-4
3.8.6	SMG\$ Access Violation—Problem Corrected	3-4
3.8.7	SMG\$ Virtual Display Access Violation—Problem Corrected	3-4
3.8.8	SMG\$CHANGE_PBD_CHARACTERISTICS Routine with DECwindows DECterm—Problem Corrected	3-4
3.8.9	SMG\$CHANGE_RENDITION Display—Problem Corrected	3-4
3.8.10	SMG\$CHANGE_VIEWPORT Label Disappearance—Problem Corrected	3-4
3.8.11	SMG\$CHANGE_VIRTUAL_DISPLAY—Problems Corrected	3-4

3.8.12	SMG\$CREATE_MENU Access Violation—Problem Corrected	3-5
3.8.13	SMG\$DISABLE_BROADCAST_TRAPPING Incompatible with SMG\$DISABLE_UNSOLICITED_INPUT—Problem Corrected	3-5
3.8.14	SMG\$DISABLE_UNSOLICITED_INPUT Invalid Channel Error—Problem Corrected	3-5
3.8.15	SMG\$EXECUTE_COMMAND Aborted by STOP/ID=xxxx—Problem Corrected	3-5
3.8.16	SMG\$INSERT_CHARS Wrap—Problem Corrected	3-5
3.8.17	SMG\$INVALIDATE_DISPLAY Wide Character Display—Problem Corrected	3-5
3.8.18	SMG\$M_RETURN_IMMED Flag—Problem Corrected	3-5
3.8.19	SMG\$PASTE_VIRTUAL_DISPLAY Screen Output—Problem Corrected	3-5
3.8.20	SMG\$PRINT_PASTEBOARD Routine—Problem Corrected	3-6
3.8.21	SMG\$PUT_CHARS_MULTI Routine Rendition Argument—Problem Corrected	3-6
3.8.22	SMG\$PUT_CHARS_WIDE and SMG\$PUT_CHARS_HIGHWIDE Text Overwritten—Problem Corrected	3-6
3.8.23	SMG\$PUT_CHARS_WIDE ASCII Characters Displayed—Problem Corrected	3-6
3.8.24	SMG\$PUT_CHARS_WIDE and SMG\$PUT_CHARS_HIGHWIDE Start-Row and Start-Column Arguments—Problem Corrected	3-6
3.8.25	SMG\$PUT_HELP_TEXT with INVISIBLE Attribute—Problem Corrected	3-6
3.8.26	SMG\$PUT_LINE Output—Problem Corrected	3-6
3.8.27	SMG\$PUT_LINE_HIGHWIDE Routine—Problem Corrected	3-7
3.8.28	SMG\$PUT_LINE_MULTI Routine—Problems Corrected	3-7
3.8.29	SMG\$PUT_STATUS_LINE Display Character—Problem Corrected	3-7
3.8.30	SMG\$READ_COMPOSED_LINE Routine—Problems Corrected	3-7
3.8.31	SMG\$READ_STRING Routine—Problems Corrected	3-8
3.8.32	SMG\$READ_VERIFY Input_Length—New Optional Argument	3-8
3.8.33	SMG\$READ_VERIFY Routine Resultant-String Argument—Problem Corrected	3-9
3.8.34	SMG\$REMOVE_LINE Line Characters—Problem Corrected	3-9
3.8.35	SMG\$SAVE_VIRTUAL_DISPLAY Special Graphics Display—Problem Corrected	3-9
3.8.36	SMG\$SNAPSHOT Routine—Problem Corrected	3-9
3.8.37	SMG\$UNPASTE_VIRTUAL_DISPLAY Return Status—Problem Corrected	3-10
3.8.38	TERMTABLE Print-Screen Definitions—Problem Corrected	3-10
3.9	SCSI Disk Class Driver Audio \$QIO Functions	3-10
3.10	System Services Notes	3-10
3.10.1	\$ADJWSL Argument—Problem Corrected	3-10
3.10.2	\$CREMBX System Service—Maximum Allowable BUFQUO Value Changed	3-10
3.10.3	Return Codes from \$GETQUI and \$\$SNDJBC—Problem Corrected	3-11
3.10.4	\$\$SNDJBC System Service—\$JC\$_CHARACTERISTIC_NUMBER Problem Corrected	3-11
3.10.5	\$\$START_TRANS—Status Return Codes Altered	3-11
3.11	VAX BASIC Run-Time Library READ REGARDLESS Clause - Problem Corrected	3-11
3.12	VAX C Run-Time Library Notes	3-11
3.12.1	Atof Function—Problem Corrected	3-12

3.12.2	Ceil and Floor Functions—Problem Corrected	3-12
3.12.3	Longjmp Function—Problem Corrected	3-12
3.12.4	Math Functions—Problem Corrected	3-12
3.12.5	Qsort Function—Problem Corrected	3-12
3.12.6	Scanw Function—Problem Corrected	3-12
3.12.7	SETVBUF Program—Problem Corrected	3-12
3.12.8	Socket Function Calls—Problems Corrected	3-12
3.12.9	Strncat Function—Problem Corrected	3-13
3.13	VAX Text Processing Utility (VAXTPU)	3-13
3.13.1	GET_INFO Built-In Procedure—New Parameter	3-13
3.13.2	READ_LINE Built-In Procedure—Problem Corrected	3-13
3.13.3	SCANL and SPANL Built-In Procedures—Problem Corrected	3-13
3.13.4	SEARCH Built-In Procedure—Problem Corrected	3-13
3.13.5	SET (SCREEN_UPDATE) Built-In Procedure—New Optional Parameter	3-13
3.13.6	SET (SCREEN_UPDATE) Built-In Procedure—Restrictions	3-14
3.13.7	New Security Option Bit Added to VAXTPU Callable Interface	3-15
3.13.8	WRITE_FILE Built-In Procedure Output—Problem Corrected	3-15
3.14	VMS Debugger	3-15
3.14.1	Debug with VMS Linker—Problem Corrected	3-15
3.14.2	Debugging Multi-Image Programs—Problem Corrected	3-15
3.14.3	Debugging Programs with LCK\$M_DEQALL Modifier – Suggested Workaround	3-16
3.15	XQP and File System	3-16
3.15.1	Erase-on-Delete Data Blocks Not Erased—Problem Corrected	3-16
3.15.2	File Truncation—Problem Corrected	3-16
3.15.3	Lock-Conversion Condition Caused System Failure—Problem Corrected	3-16
3.15.4	Null-Mode Lock Request Caused System Failure—Problem Corrected	3-16
3.15.5	SS\$_DEADLOCK Error Status Returned on File Creation or Extension—Problem Corrected	3-16

4 Documentation Release Notes

4.1	<i>VMS Audit Analysis Utility Manual</i>	4-1
4.2	<i>VMS Device Support Reference Manual</i>	4-1
4.3	<i>VMS Network Control Program Manual</i>	4-1
4.4	<i>VMS System Services Reference Manual</i>	4-2
4.5	<i>VMS Version 5.5 Upgrade and Installation Manual</i>	4-2

A Audio Extensions to the SCSI Disk Class Driver

A.1	SCSI Audio Commands	A-1
A.2	\$QIO Interface to Audio Functionality of the SCSI Disk Class Driver	A-2
A.3	Defining an Audio Control Block (AUCB)	A-3
A.4	Error Handling in Applications Using SCSI Audio Functions	A-6
A.5	Using CD-ROM to Store Both Data and Audio Information	A-8
A.6	Programming Audio Applications	A-8
A.7	Application Program Example Using SCSI Audio Capabilities	A-9

B New and Changed System Messages

C VMS TURBOchannel Device Driver Support

C.1	Hardware Environment	C-1
C.1.1	Address Maps	C-2
C.2	DMA Transactions	C-3
C.2.1	4 MB Mapped DMA Space	C-4
C.2.2	104 MB Unmapped DMA Space	C-4
C.2.3	Using TURBOchannel DMA Routines	C-4
C.3	Handling Interrupts	C-5
C.4	Coding a TURBOchannel Device Driver	C-5
C.4.1	SYSGEN Autoconfigure	C-6
C.5	Assembling and Linking a TURBOchannel Driver	C-6
C.6	Loading a TURBOchannel Device Driver	C-7
C.7	TURBOchannel Operating System Routines	C-8
	IOC\$ALOTCMAP_DMA, IOC\$ALOTCMAP_DMAN	C-9
	IOC\$LOADTCMAP_DMA, IOC\$LOADTCMAP_DMAN	C-11
	IOC\$RELTCMAP_DMA, IOC\$RELTCMAP_DMAN	C-13

Index

Figures

2-1	Serving Nodes in a CI or DSSI Configuration	2-41
A-1	Audio Control Block (AUCB)	A-3
A-2	Output Channel Selection and Volume Settings for CD-ROM Ports as Used by the SET_VOLUME Function	A-6
C-1	VAXstation with a TURBOchannel Subsystem	C-2
C-2	TURBOchannel Map for the VAXstation CPU	C-2
C-3	TURBOchannel DMA to a VAX Host	C-3
C-4	TURBOchannel Map Register	C-5
C-5	TURBOchannel Map Register Descriptor (TC_MD)	C-10

Tables

2-1	VAX Boot Commands	2-23
2-2	VAX Boot Command Qualifier Values	2-23
2-3	Controllers Supporting Copy and Merge Assists	2-33
2-4	Copy Times in Minutes for Two-Member RA82 Shadow Sets	2-35
3-1	Possible Screen Effects of [window] Parameter	3-14
4-1	License Unit Requirement Table (LURT)	4-2
A-1	SCSI Disk Class Driver Audio Commands	A-1
A-2	Contents of Audio Control Block	A-4
A-3	VMS Status Codes Returned to the IOSB and AUCB by the SCSI Disk Class Driver	A-7
C-1	Driver Entry Point Routines	C-6

Preface

This document describes software problems, corrections, restrictions, documentation changes, and new hardware support that pertain to Version 5.5–2 of the VMS operating system.

To apply the Version 5.5–2 update, you must be running at least VMS V5.5 or VMS A5.5 on your system.

Note

The numerals that appear in the left margin beneath the title of every release note indicate the version number of the VMS operating system to which that release note applies. For instance, a release note that appears with 5.5-1 in the left margin indicates that the information in this note was effective starting with VMS Version 5.5–1.

Intended Audience

The *VMS Version 5.5–2 Release Notes* are intended for all system users. Read the release notes before applying the Version 5.5–2 update.

Document Structure

This manual contains the following chapters and appendixes:

- Chapter 1 contains release notes intended for general users of the VMS operating system.
- Chapter 2 contains release notes intended for system managers.
- Chapter 3 contains release notes intended for application and system programmers.
- Chapter 4 contains additions and corrections to the VMS Documentation Set.
- Appendix A contains information on audio extensions to the disk class driver.
- Appendix B contains new system messages.
- Appendix C contains information on the VMS TURBOchannel device driver.

Conventions

The following conventions are used in this manual:

Ctrl/ <i>x</i>	A sequence such as Ctrl/ <i>x</i> indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.
PF1 <i>x</i>	A sequence such as PF1 <i>x</i> indicates that you must first press and release the key labeled PF1, then press and release another key or a pointing device button.
Return	In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)
...	A horizontal ellipsis in examples indicates one of the following possibilities: <ul style="list-style-type: none">• Additional optional arguments in a statement have been omitted.• The preceding item or items can be repeated one or more times.• Additional parameters, values, or other information can be entered.
.	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.
()	In format descriptions, parentheses indicate that, if you choose more than one option, you must enclose the choices in parentheses.
[]	In format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in a VMS file specification, or in the syntax of a substring specification in an assignment statement.)
{ }	In format descriptions, braces surround a required choice of options; you must choose one of the options listed.
boldface text	<p>Boldface text represents the introduction of a new term or the name of an argument, an attribute, or a reason.</p> <p>Boldface text is also used to show user input in online versions of the manual.</p>
<i>italic text</i>	Italic text emphasizes important information, indicates variables, and indicates complete titles of manuals. Italic text also represents information that can vary in system messages (for example, Internal error <i>number</i>), command lines (for example, /PRODUCER= <i>name</i>), and command parameters in text.
UPPERCASE TEXT	Uppercase text indicates a command, the name of a routine, the name of a file, the name of a file protection code, or the abbreviation for a system privilege.
-	A hyphen in code examples indicates that additional arguments to the request are provided on the line that follows.

numbers

All numbers in text are assumed to be decimal, unless otherwise noted. Nondecimal radices—binary, octal, or hexadecimal—are explicitly indicated.

General User Release Notes

This chapter contains information about the VMS Version 5.5-2 operating system for the general user.

1.1 Digital Command Language (DCL)

The release notes in this section pertain to commands used in the DIGITAL Command Language (DCL).

1.1.1 EDIT/TPU Command /NOWORK Qualifier—Problem Corrected

V5.5-2 In previous versions of VMS, a problem existed with the /NOWORK qualifier for the DCL command EDIT/TPU. This qualifier did not disable the creation and use of a work file. In VMS Version 5.5-2, this problem has been corrected.

1.1.2 Error Message Displayed When Non-Modem Port Is Set to Modem

V5.5-2 In VMS Version 5.5-2, the TTDRIVER has been modified to report an error if you try to set a port using either the SET TERMINAL/MODEM or the SET TERMINAL/COMMSYNC command, and that port does not have hardware modem signal support. This modification has no effect if the port has hardware modem signal support.

The following is an example of the error message:

```
%SET-W-NOTSET, error modifying TWA1:
-SYSTEM-E-UNSUPPORTED, unsupported operation or function
```

In previous versions of VMS, if you issued the command SET TERMINAL /MODEM on a port without hardware modem signal support, the system ignored the command and sent no message.

1.1.3 ON Condition Response—Problem Corrected

V5.5-2 In previous versions of VMS, DCL did not respond with the ON condition when there was a warning or an error in a command procedure invoked by the last line of another command procedure. The following example illustrates this problem:

```
$ TYPE MYFILE.COM
$ !MYFILE.COM
$ ON WARNING THEN WRITE SY$OUTPUT "WARNING CONDITION DELIVERED"
$ OPEN MYFILE1 MYFILE1.COM/WRITE
$ WRITE MYFILE1 "$EXIT 0 ! RETURN ERROR"
$ CLOSE MYFILE1
$
$ @MYFILE1.COM !this is the last line of this command procedure
$
$
$@MYFILE.COM
%NONAME-W-NOMSG, Message number 00000000
```

Note that in this example, VMS does not deliver the ON WARNING condition.

General User Release Notes

1.1 Digital Command Language (DCL)

In VMS Version 5.5-2, this problem has been corrected. DCL now correctly delivers the ON condition when an error occurs because of a command procedure invoked by another command procedure.

1.1.4 READ Command Error Message

V5.5-2 In VMS Version 5.5-2, the DCL command READ has been changed to provide an error message if you specify the /NOPROMPT qualifier. In previous versions of VMS, DCL ignored the /NOPROMPT qualifier, and returned no error message. In VMS Version 5.5-2, using the /NOPROMPT qualifier with the READ command results in the following error message:

```
%DCL-W-NOTNEG qualifier or keyword not negatable - remove "NO" or omit
\nOPROMPT\
```

1.1.5 SET DEFAULT and SHOW DEFAULT Commands—Behavior Changed

V5.5-2 In VMS Version 5.5-2, the DCL command SET DEFAULT does not set the default correctly when you use a logical name that has multiple translations. For example:

```
$ SET DEFAULT DISK1:[DIRECTORY1]
$ DEFINE myfile myfile1
$ DEFINE myfile1 DISK2:[DIRECTORY2]
$ SET DEFAULT myfile
$ SHOW DEFAULT
  DISK1:[DIRECTORY2]
```

In this example, the default should read:

```
DISK2:[DIRECTORY2]
```

This problem will be corrected in a future release of VMS.

Currently, DCL does not show the right default device in the SHOW DEFAULT command when you use a logical name that has multiple translations. For example:

```
$ SET DEFAULT DISK1:[DIRECTORY]
$ DEFINE myfile myfile1
$ DEFINE myfile1 DISK2:[DIRECTORY]
$ SET DEFAULT myfile: ! Must be with a colon
$ SHOW DEFAULT
  myfile:[DIRECTORY]
$ WRITE SYS$OUTPUT F$ENVIRONMENT("DEFAULT")
  myfile:[DIRECTORY]
```

In this example, "myfile" is used as the default device. The correct default directory display should be DISK2:[DIRECTORY].

This SHOW DEFAULT behavior will be changed in a future release of VMS.

1.1.6 SET MESSAGE Command Changed—Problem Corrected

V5.5-2 In previous versions of VMS, using the DCL command SET MESSAGE opened the message file in kernel mode. This sometimes caused deadlocks to occur in the rundown of processes if too many processes accessed the same message file simultaneously. If this deadlock occurred, a process could not complete the logout process and you had to stop the process manually.

In VMS Version 5.5-2, this problem has been corrected. The image invoked by the DCL command SET MESSAGE has been changed so that the message file is not opened in kernel mode. This change to SET MESSAGE prevents any deadlocks from happening in this situation.

General User Release Notes 1.1 Digital Command Language (DCL)

1.1.7 SET TERMINAL/SPEED Command—Problem Corrected

V5.5-2 In previous versions of VMS, the DCL command SET TERMINAL /SPEED=38400 on TTxx type ports improperly set the port baud rate to 110. In VMS Version 5.5-2, this problem has been corrected in the YEDRIVER; the command no longer has any effect on the current baud rate. The TTxx hardware type devices do not support the 38400 baud rate so setting this baud rate would cause other problems. The hardware devices that do support 38400 baud rate will not be affected by this change. The DCL command SET TERMINAL/SPEED=38400 on these devices will work the same as in the past.

1.1.8 SET TERMINAL/COMMSYNC Command—Support Added

V5.5-2 VMS Version 5.5-2 adds modified modem polling support to the YEDRIVER for the DCL command SET TERMINAL/COMMSYNC. The TTAAn port requires modified modem polling that is not required by other terminal ports, such as TXAn ports.

1.1.9 SHOW DEVICE/FULL Command—Problem Corrected

V5.5-2 In previous versions of VMS, the DCL command SHOW DEVICE/FULL stopped displaying device information when it encountered a device with access control list entries that denied access. Notice that in the following example, when access to device DUA0 was restricted, the SHOW DEVICE/FULL command did not display information for device DUA1.

```

$ SHOW DEVICE DU
Device      Device      Error      Volume      Free      Trans Mnt
Name        Status      Count      Label       Blocks   Count  Cnt
DUA0:      Mounted     0          TEST        145734    1      1
DUA1:      Mounted     0          VAXVMSRL055 46062    262    1

$ SHOW PROCESS

6-MAR-1992 17:30:07.66  User: BAR          Process ID: 00000A68
                        Node: FOO          Process name: "BAR"

:
:

$ SET PROCESS/PRIVILEGE=ALL
$ SET ACL/ACL=(IDENTIFIER=[BAR],ACCESS=NONE)/OBJECT=DEVICE DUA0:
$ SET PROCESS/PRIVILEGE=NOALL
$ SHOW DEVICE/FULL DU
Disk FOO::DUA0:, device type .....

Error count          0  Operations completed          177041
Owner process        ""  Owner UIC                      [300,240]
Owner process ID    00000000  Dev Prot  S:RWED,O:RWED,G:RWED,W:RWED
Reference count     1  Default buffer size            512
Total blocks        204864  Sectors per track              33
Total cylinders     776  Tracks per cylinder             8

Volume label        "TEST"  Relative volume number          0
Cluster size        3  Transaction count              1
Free blocks         145734  Maximum files allowed          30000
Extend quantity     5  Mount count                    1
Mount status        System  Cache name                      " FOO$DUA0:XQPCACHE"
Extent cache size   64  Maximum blocks in extent cache 14573
File ID cache size  64  Blocks currently in extent cache 75
Quota cache size    0  Maximum buffers in FCP cache    206

Device access control list:
%SYSTEM-F-NOPRIV, no privilege for attempted operation

```

General User Release Notes

1.1 Digital Command Language (DCL)

In VMS Version 5.5-2, this problem has been corrected. SHOW DEVICE/FULL continues displaying information of other devices that are available on the system.

1.1.10 TYPE Command /PAGE Qualifier—Enhanced Screen Usage

V5.5-2 In VMS Version 5.5-2, the /PAGE qualifier for the DCL command TYPE has been enhanced to better use the screen. In previous versions of VMS, lines that were exactly as wide as the screen were counted as two lines. In VMS Version 5.5-2, as much of the screen as possible is filled.

1.2 Mail Utility (MAIL)

The following release notes pertain to the VMS Mail Utility (MAIL).

1.2.1 MAIL Sends and Receives Compound DDIF and DTIF Documents

V5.5-2 In VMS Version 5.5-2, MAIL sends and receives compound DDIF or DTIF documents. MAIL uses these documents in a manner similar to DECwindows Mail.

When MAIL receives a DDIF or DTIF document, it prompts the user to extract the message to an external file. The EXTRACT command creates the DDIF or DTIF file (and all related files).

1.2.2 Missing Error Message in MAIL—Problem Corrected

V5.5-2 In previous versions of VMS, when you used a TPU-based callable editor (such as EVE or LSE) with the VMS Mail Utility, and the terminal was not an ANSI-compliant CRT, the requested edit did not occur, no error message showed, and the next MAIL> prompt appeared.

In VMS Version 5.5-2, this problem has been corrected. The following error message appears for these conditions:

```
%TPU-E-NONANSICRT, SYS$INPUT must be supported CRT
```

1.3 VAXstation 4000 Keyboard Problem

V5.5-2 When you use a VAXstation 4000 series machine, and you also use the graphics head console, pressing the HALT button and entering CONTINUE at the console prompt leaves the keyboard in a reset state. In this state, the keyboard sometimes autorepeats (duplicate letters appear). To fix the keyboard state, unplug the keyboard, then plug it in again.

1.4 VAX Text Processing Utility (VAXTPU)

The release notes in this section pertain to changes and corrections to the VAX Text Processing Utility (VAXTPU) and the EVE editing application within VAXTPU.

1.4.1 EVE Key Name Definitions—Problem Corrected

V5.5-2 In previous versions of VMS, when the EDT keypad was active, the EVE command SHOW KEY returned invalid key names for the GOLD-0 through GOLD-9 keys. This problem affected the display generated by the HELP KEYS command for those keys under the column labeled "Gold sequence." If you entered HELP KEYS, EVE displayed the following information (where <LF> is the line feed character and <CR> is a carriage return):

General User Release Notes

1.4 VAX Text Processing Utility (VAXTPU)

```
      Gold sequence
      Function
(GOLD = PF1)
.      .
.      .
.      .
GOLD-1.0      Repeat
GOLD-<LF>SINGLE_QUO      Repeat
GOLD-<LF>GOLD<LF><CR>SHIF      Repeat
GOLD-GOLD      Repeat
GOLD-FUNCTION      Repeat
GOLD-KEYPAD      Repeat
GOLD-SHIFT      Repeat
GOLD-CTRL      Repeat
GOLD-HELP      Repeat
GOLD-ALT      Repeat
.      .
.      .
```

In VMS Version 5.5-2, EVE displays the correct key names GOLD-0 through GOLD-9.

1.4.2 FILL Built-In Procedure—Problems Corrected

V5.5-2 In previous versions of VMS, a problem existed in which the VAXTPU built-in procedure FILL affected characters at the start of a range. This problem occurred when you filled a multiline range and the beginning of the range was not the beginning of a line. VAXTPU changed the position of the text at the beginning of the range, inserting a line break. In VMS Version 5.5-2, this problem has been corrected.

Also in previous versions of VMS, a similar problem existed in which the VAXTPU built-in procedure FILL split a line at an incorrect point when you filled a 1-line range. In VMS Version 5.5-2, this problem has been corrected.

1.4.3 Tabs in Overstrike Mode—Problem Corrected

V5.5-2 In previous versions of VMS, when you entered tabs using overstrike mode (by typing over existing characters rather than inserting them between these characters), buffer change journaling occasionally caused an access violation error. In VMS Version 5.5-2, this problem has been corrected.

1.4.4 Work Files—Problem Corrected

V5.5-2 In previous versions of VMS, VAXTPU work files that consisted of temporary files were deleted when you exited VAXTPU. You could not access these work files.

In VMS Version 5.5-2, the work files are accessible when you exit from VAXTPU. These files reside in SYS\$SCRATCH:TPU\$WORK.TPU\$WORK.

System Manager Release Notes

This chapter includes information about the VMS Version 5.5-1 and VMS Version 5.5-2 operating systems for system managers.

For a list of relevant system messages that are new in VMS Version 5.5-1 and in VMS Version 5.5-2, refer to Appendix B.

2.1 Allocation Classes in VAXcluster Configurations—Restriction

V5.5-2 For VMS Version 5.5-2, you cannot change allocation classes when the VAXcluster system is running. Changing allocation classes causes the affected device(s) to have multiple device names across the cluster. This can cause data corruption. To prevent the possibility of data corruption, VMS Version 5.5-2 will force a system failure if it detects a change in controller allocation class.

Changes can be performed only during regularly scheduled VAXcluster downtime for maintenance and upgrades. For more information, see the *VMS VAXcluster Manual*.

2.2 Audit Analysis Utility /SELECT=STATUS=FAILURE Qualifier—Problem Corrected

V5.5-2 In previous versions of VMS, a problem existed in the VMS Audit Analysis Utility that caused the system to refuse to accept the /SELECT=STATUS=FAILURE qualifier. In VMS Version 5.5-2, this problem has been corrected. The system now accepts the /SELECT=STATUS=FAILURE qualifier.

2.3 AUTOGEN Command Procedure Notes

The notes in this section pertain to the AUTOGEN command procedure.

2.3.1 AUTOGEN Problem When Executed from SYSMAN

V5.5-2 When you execute AUTOGEN on a remote node using the System Management Utility (SYSMAN), AUTOGEN might return an access violation error during the GETDATA phase. For example, an access violation error might occur if you enter the following commands:

```
$ RUN SYSSYSTEM:SYSMAN
SYSMAN> SET ENVIRONMENT/NODE=(NODE1,NODE2,NODE3)
SYSMAN> DO @SYSSUPDATE:AUTOGEN SAVPARAMS SETPARAMS
```

Digital suggests you execute AUTOGEN using a batch-oriented command procedure as explained in the *Guide to Setting Up a VMS System*. If the command procedure is not practical for an individual case, log in to the desired node and execute AUTOGEN interactively.

System Manager Release Notes

2.3 AUTOGEN Command Procedure Notes

2.3.2 Specifying a Minimum Required Age for AUTOGEN Feedback Data

V5.5-2 With AUTOGEN, feedback data is useful only when a system has been running long enough to accurately reflect the system's normal workload. By default, AUTOGEN uses feedback data if the data is older than 24 hours. You can define the logical name AGEN\$FEEDBACK_REQ_TIME to specify, in hours, a different minimum age required for feedback data. AUTOGEN uses this value to determine whether to use the feedback data.

For example, you might define the logical name as shown, to indicate that AUTOGEN should use feedback data if it is older than 19 hours. Add this line in SYLOGICALS.COM so it can be defined each time the system boots:

```
$ DEFINE AGEN$FEEDBACK_REQ_TIME 19
```

When AUTOGEN runs, it displays whether feedback data is used, as follows:

```
Feedback information was collected on 21-JUN-1992 14:00:08.53
Old values below are the parameter values at the time of collection.
The feedback data is based on 19 hours of up time.
Feedback information will be used in the subsequent calculations
```

For more information on using AUTOGEN and feedback data, see the *Guide to Setting Up a VMS System*.

2.3.3 Specifying the Number of Ethernet Adapters for AUTOGEN

V5.5-2 In a VAXcluster environment, you can define the symbol NUM_ETHERADAPT in the file SYS\$SYSTEM:MODPARAMS.DAT to specify the total number of Ethernet adapters in the cluster.

For example, you might include the following line in MODPARAMS.DAT:

```
NUM_ETHERADAPT = 40
```

For more information on using AUTOGEN and defining symbols in MODPARAMS.DAT, see the *Guide to Setting Up a VMS System*.

2.3.4 Specifying the Number of VAXcluster Nodes for AUTOGEN

V5.5-2 You can define the symbol NUM_NODES in the file SYS\$SYSTEM:MODPARAMS.DAT to specify the number of nodes that are to run in a VAXcluster system. AUTOGEN uses this value to set parameters that are affected by the number of VAXcluster nodes.

For example, you might include the following line in MODPARAMS.DAT:

```
NUM_NODES = 30
```

For more information on using AUTOGEN and defining symbols in MODPARAMS.DAT, see the *Guide to Setting Up a VMS System*.

2.3.5 Suggested Workaround to SYSMAN Value Error

V5.5-1 If you run AUTOGEN through the SETPARAMS phase, it may display the following error:

```
%SMI-E-OUTRANGE, parameter is out of range
%AUTOGEN-I-ERROR, SETPARAMS phase was aborted due to
an unexpected error.
```

System Manager Release Notes

2.3 AUTOGEN Command Procedure Notes

This error occurs when one of the parameters is above the maximum or below the minimum value allowed for that parameter. When AUTOGEN invokes the System Management Utility (SYSMAN) to set the parameter value, SYSMAN corrects the problem. For example, if a parameter is to be set above its maximum value, SYSMAN sets the parameter at the maximum value and continues. When SYSMAN completes, AUTOGEN detects that an error has occurred and displays the abort message. If you review the parameter values set by AUTOGEN, you will see that they have been set correctly.

To identify which parameter is causing the problem, create a command procedure similar to the following one and use your editor to include the file SYS\$SYSTEM:SETPARAMS.DAT:

```
$!  
$ SET VERIFY  
$!  
$ RUN SYS$SYSTEM:SYSMAN  
$!  
$! Include SYS$SYSTEM:SETPARAMS.DAT  
$!  
$ EXIT
```

When you execute the command procedure, the parameter names will be displayed, allowing you to find the out-of-range parameter.

To fix the cause of the problem, define the maximum or minimum value for the parameter in the AUTOGEN parameter file SYS\$SYSTEM:MODPARAMS.DAT by using the MAX_ or MIN_ prefix as follows:

```
MIN_parameter-name = minimum-value  
MAX_parameter-name = maximum-value
```

For example, to specify a maximum value of 400000 for PAGEDYN, add the following line to SYS\$SYSTEM:MODPARAMS.DAT:

```
MAX_PAGEDYN = 400000
```

For information about the MAX_ and MIN_ prefixes and MODPARAMS.DAT, see the *Guide to Setting Up a VMS System*.

2.3.6 TMSCP_LOAD Parameter—Problem Corrected

V5.5-2

VMS Version 5.5 introduced the system parameter TMSCP_LOAD to allow loading of the tape mass storage control protocol (TMSCP) server software. However, a problem existed in Version 5.5 causing AUTOGEN to inadvertently set the value of this parameter to 1. Setting this parameter to 1 loads the TMSCP software and sets locally connected tapes served. This problem was documented in the VMS Version 5.5 cover letter.

With VMS Version 5.5-2, this problem has been corrected. The value of TMSCP_LOAD is now 0 unless you explicitly change it either manually or by specifying the parameter value in the AUTOGEN parameter file SYS\$SYSTEM:MODPARAMS.DAT.

Check the file MODPARAMS.DAT to make sure AUTOGEN correctly handles this parameter for your system, as follows:

- If you find the following line in MODPARAMS.DAT, remove it:

```
TMSCP_LOAD = 0
```

This line was added to work around the AUTOGEN problem; it is no longer required, because 0 is the default value.

System Manager Release Notes

2.3 AUTOGEN Command Procedure Notes

- If you want to use the TMSCP server software, include the following line in MODPARAMS.DAT:

```
TMSCP_LOAD = 1
```

This line directs AUTOGEN to set the value of this parameter to 1.

2.4 Backup Utility (BACKUP)

The release notes in this section pertain to the VMS Backup Utility (BACKUP).

2.4.1 BACKUP Denies Privileged Users Access to a Tape

V5.5-2

In VMS Version 5.4–3, the fix of the problem to BACKUP that corrected the use of tapes with protection masks was incomplete. In certain situations, users with the correct privileges are unable to access a tape. You can add the protection mask to the volume with the DCL command INITIALIZE prior to using the tape for a BACKUP save operation, or you can specify a protection mask on the BACKUP command line with the /PROTECTION qualifier. For example, a tape may have the following protection mask:

```
SYSTEM:READ,WRITE and OWNER:READ,WRITE
```

If a tape has this protection mask and is owned by UIC [1,4], a user who has a UIC of [100,77] and who is fully privileged cannot access the tape. You can mount the tape, but BACKUP denies you use of the tape.

2.4.2 BACKUP/LIST Operations on a Save Set

V5.5-2

In previous versions of the VMS operating system, there was a problem with using a wildcard character in the directory specification of a BACKUP/LIST command. BACKUP would do one of the following:

- Exit normally if it could not find a save set on the disk that matched the input save set
- Loop endlessly, displaying the first save set on the disk that matched the input save set

With VMS Version 5.5–2, BACKUP correctly searches all directories that match the wildcard specification and lists all matching save sets.

2.4.3 Data Compactions on Tape Drives

V5.5-2

The *VMS Version 5.5 Release Notes* recommended extending the BACKUP XOR error correction for data compaction drives. However, because of the high reliability of data compaction tape drives introduced with VMS Version 5.3–1, this is no longer necessary, and you no longer need to extend the BACKUP XOR error correction.

2.4.4 Extending Index Files

V5.5-2

In previous versions of the VMS operating system, a problem occurred when you initialized a disk and extended the INDEXF.SYS file using the /HEADERS and /MAXIMUM_FILES qualifiers. During a BACKUP/IMAGE restore or copy operation with /NOINITIALIZE as the output-device qualifier, BACKUP ignored the extension that you made to the index file. BACKUP created on the target disk an INDEXF.SYS file that was the same size as the INDEXF.SYS file on the source disk.

System Manager Release Notes

2.4 Backup Utility (BACKUP)

With VMS Version 5.5-2, BACKUP preserves the size of the INDEXF.SYS file on the target disk when you use /NOINITIALIZE as an output-device qualifier. As a result, the index file on the target disk must be large enough to accommodate the number of files that will be copied from the source disk. If INDEXF.SYS is too small, the BACKUP operation aborts and displays an error message.

2.4.5 Message Status Codes Restored

V5.5-2 In previous versions of VMS, the numeric values for BACKUP message status codes could change with each new release of VMS. This sometimes caused a user-written command procedure to fail if it explicitly specified the numeric values for status code checks.

VMS Version 5.5-2 restores the values for BACKUP message status codes that existed since VMS Version 5.5. Future versions of the VMS operating system will not redefine the existing values for BACKUP message status codes.

2.4.6 Multivolume Logical Names

V5.5-2 In previous versions of the VMS operating system, when BACKUP dismounted the first tape drive of a multivolume save or restore, it deassigned the logical name that was defined for the device when it was mounted.

With VMS Version 5.5-2, BACKUP preserves the first tape drive of a multivolume save or restore, it deassigned the logical name that was defined for the device when it was mounted.

2.4.7 Multivolume Save Operations from a Subprocess

V5.5-2 In previous versions of the VMS operating system, you could not complete a multivolume save operation from a subprocess when you used BACKUP. The operation failed and issued an error message stating that the tape drive was allocated to another user (the parent process).

With VMS Version 5.5-2, BACKUP will handle a multivolume backup from a subprocess by explicitly allocating the tape drive to the subprocess. If the tape drive is allocated to the parent process when you start your backup in the subprocess, BACKUP fails with the following message:

Device already allocated.

2.4.8 Saving and Restoring Alias Directories

V5.5-2 On a VMS system disk, the file SYSCOMMON.DIR is an alias directory of the file [000000]VMS\$COMMON.DIR. This means that both files point to the same file header. In previous versions of the VMS operating system, this caused a problem performing an image backup and restore of the system disk. During the restore operation, BACKUP did not properly restore the VMS\$COMMON.DIR file. Although this does not affect the system disk, it might produce errors with DIGITAL Command Language (DCL) lexical functions.

To determine if your system disk has this problem, make sure that you have the LOGIO privilege and enter the DUMP/HEADER/BLOCK command as follows:

System Manager Release Notes

2.4 Backup Utility (BACKUP)

```
$ DUMP/HEADER/BLOCK=(COUNT=0) DISK:[000000]VMS$COMMON.DIR
  Map area offset:          100
  Access control area offset: 255
  Reserved area offset:     255
.
.
.
Identification area
  File name:                SYSCOMMON.DIR;1
  Revision number:          3
  Creation date:            15-JUN-1989 05:27:35.68
  Revision date:            23-JUN-1992 13:13:53.04
  Expiration date:         <none specified>
  Backup date:              <none specified>

Map area
  Retrieval pointers
    Count:      2      LBN:      5411

Checksum:      16366
```

If the file name in the Identification Area part of the display is not VMS\$COMMON.DIR, as shown in this example, your system disk is affected by this problem. To restore VMS\$COMMON to its proper state, enter the following commands:

```
$ SET FILE/ENTER=SYSCOMMON.DIR VMS$COMMON.DIR
$ SET FILE/REMOVE VMS$COMMON.DIR
$ RENAME SYSCOMMON.DIR VMS$COMMON.DIR
```

If you execute the DUMP command again, the Identification Area part of the display should contain VMS\$COMMON.DIR.

With VMS Version 5.5-2, BACKUP handles alias directories correctly during an image restore operation.

Note

Restoring image backups created with a previous VMS version will cause the problem to recur.

2.4.9 Tape Volume Protection

V5.5-2

In previous versions of the VMS operating system, initializing a tape volume with the INITIALIZE/PROTECTION command and then using the tape in a BACKUP/REWIND save operation had the following effects:

- BACKUP changed the protection, giving world write access.
- Users could initialize the tape regardless of whether they had write access to the tape.

With VMS Version 5.5-2, BACKUP correctly handles the protection of a tape volume in this situation.

2.5 Batch and Print Queuing System

The following release notes pertain to the VMS Batch and Print Queuing System.

System Manager Release Notes 2.5 Batch and Print Queuing System

2.5.1 ALL-IN-1 and Batch and Print Queuing System—Interaction Problems Corrected

V5.5-1 In VMS Version 5.5, there were interaction problems between the new Batch and Print Queuing System and the ALL-IN-1 software that caused the OA\$FORMATTER queue to function improperly and occasionally caused the queue manager process to abort.

In VMS Version 5.5-1, these problems have been corrected.

2.5.2 Chance of Queuing System Shutting Off During Node Shutdown

V5.5-2 In VMS Version 5.5, there was a small chance that the Batch and Print Queuing System would turn off when you shut down a standalone machine, or from a node on which you removed the queue manager from a cluster. In VMS Version 5.5-2, this problem has been corrected.

Be aware that there are other events during the shutdown procedure that can cause the queue manager to fail. For example, if you keep the queuing database on a disk other than the system disk, and then dismount that disk in the shutdown procedure, it could cause the queue manager to fail. Another example is in a cluster environment if you shut down a node after the queue manager fails over to that node. In this case, it is possible that the shutdown procedure will purge installed images that the queue manager needs to start up, causing the queue manager to fail.

If situations such as these occur and you bring the queue manager back up on the same node, it will certainly fail again. The queuing system is programmed to be shut down completely if the queue manager fails twice on the same node within 2 minutes. Therefore, the queuing system in this situation would have been shut off and requires a manual restart to resume queuing activity for the rest of the cluster. A standalone machine requires a manual restart of the queuing system after you reboot the system.

2.5.3 Corruption Detection for Queues

V5.5-1 The Queue Manager facility attempts to correct any kind of corruption it detects. If it detects corruption in a queue record, it may disable a queue to isolate the corruption. When a queue is disabled, the following message is written to the console and in the operator log file:

```
%QMAN-I-QUEDISCOR, "queue_name"  
has been disabled due to database corruption
```

When a queue is disabled, any attempt to modify or submit to it will return the following message:

```
%JBC-E-QUEDISABLED, disabled queue cannot be modified,  
nor can a job be submitted to it
```

If you see either of these messages, do the following:

1. Submit a Software Performance Report (SPR) and include copies of the queue file and master file of the queue database.
2. Delete the queue and create a new queue to replace it.

Specific instructions are provided with the documentation of the new system messages in Appendix B.

System Manager Release Notes

2.5 Batch and Print Queuing System

2.5.4 DEFINE/CHARACTERISTIC Command Allows Only One Name to a Number

V5.5-2 In versions of VMS prior to Version 5.5-2, the DEFINE/CHARACTERISTIC command for the Batch and Print Queuing System allowed you to define more than one characteristic name to a number, although this capability was unsupported. The result of defining multiple characteristic names to a number was unpredictable.

In VMS Version 5.5-2, the DEFINE/CHARACTERISTIC command no longer allows you to define more than one characteristic name to a number.

If your queue configuration requires you to have more than one characteristic name for a single number, you can define logical names to achieve the same result. For example, you might enter the following commands:

```
$ DEFINE/CHARACTERISTIC SECOND_FLOOR 2
$ DEFINE/SYSTEM/EXECUTIVE_MODE SALES_FLOOR SECOND_FLOOR
$ DEFINE/SYSTEM/EXECUTIVE_MODE SALES_DEPT SECOND_FLOOR
```

In this example, the characteristic name SECOND_FLOOR is assigned to the characteristic number 2. The logical names SALES_FLOOR and SALES_DEPT are then defined as equivalent to the characteristic name SECOND_FLOOR. As a result, the logical names SALES_FLOOR or SALES_DEPT are equivalent to the characteristic name SECOND_FLOOR and the characteristic number 2. These logical names can be specified as the *characteristic-name* value for any /CHARACTERISTIC=*characteristic-name* qualifier.

In a VAXcluster environment, you must define the logical names on every node that requires them.

For more information on the DEFINE/CHARACTERISTIC command, see the *VMS DCL Dictionary*. For information on using characteristics with queues, see the *Guide to Maintaining a VMS System*.

2.5.5 VAX Distributed Queuing Service (DQS) Print Symbiont—Suggested Workaround

V5.5-1 Under certain conditions, a problem exists with the VAX Distributed Queuing Service (DQS) Version 1.2 and VMS Versions V5.5 and V5.5-1, and 5.5-2.

If DQS is installed as a client only, no problem exists. However, print queues might fail if DQS is installed as a server and you have initialized any print queues with the DQS print symbiont as the processor (INITIALIZE/QUEUE/PROCESSOR=DQS\$PRTSMB.EXE).

If the print queues fail, perform the following steps:

1. If the system is hanging because the DQS print symbiont is looping, manually stop the DQS print symbiont process that is using the DQS\$PRTSMB.EXE image, as follows:
 - a. Enter the SHOW SYSTEM command to show the DQS print symbiont process that is looping. The process state is COM. The process name is SYMBIONT_xxx. Take note of the process identifier (pid); you will use this hexadecimal value to stop the symbiont process.
 - b. Enter the STOP/ID=*pid* command to stop the DQS print symbiont process. Stopping the process will clear the hanging condition. You do not need to reboot your system.

System Manager Release Notes

2.5 Batch and Print Queuing System

2. Replace the DQS print symbiont with the standard VMS print symbiont as follows:
 - a. Enter the `STOP/QUEUE/NEXT` command and stop all queues that use the `DQS$PRTSMB` image as a processor.
 - b. Enter the `START/QUEUE/PROCESSOR=PRTSMB` command for all queues that use the `DQS$PRTSMB` image as a processor. You can find which queues are using the `DQS$PRTSMB` image by specifying the `SHOW QUEUE/FULL` command. Check the value of the `/PROCESSOR` qualifier in the resulting display.

You do not need to reboot your system to reinitialize your queues.

If you perform these steps, DQS continues to function using the VMS print symbiont, with the following exceptions:

- Flag, burst, and trailer pages will not display the node and name of the remote user.
- Flag, burst, and trailer pages will display the local job number rather than the remote job number.
- The header line and the flag, burst, and trailer pages will display the time that the local job was entered rather than the entry time of the remote job.

2.5.6 Generic Queues with Implicit Target Lists—Change in Behavior

V5.5-1

A generic queue has an implicit target list if the queue is created with the `INITIALIZE/QUEUE/GENERIC` command and no queues are listed as values for the `/GENERIC` qualifier. (Although you can create a generic queue with an implicit list, you should normally specify explicit target lists for generic queues so that you can control where jobs execute.)

In VMS Version 5.4, queues with implicit target lists considered only similar queues as potential targets. That is, a generic terminal, server, printer, or batch queue could feed only a like execution queue.

In VMS Version 5.5, this behavior changed. Now, generic batch queues can still feed only batch execution queues, but any other type of generic queue can feed any symbiont execution queue (terminal, server, or printer).

The original behavior will be restored in a future release. Until then, this change in behavior might cause a problem if you use layered products that create server queues or devices that create terminal queues. To work around the problem, do one of the following:

- Explicitly specify target queues for your generic queues. To do so, specify the target queues with the `/GENERIC` qualifier for the `START/QUEUE` or `INITIALIZE/QUEUE` command in the following format:

```
INITIALIZE/QUEUE/GENERIC=(target-queue-name[,...]) queue-name[:]
```

- Set the `NOENABLE_GENERIC` flag for those execution queues that should *not* receive jobs from the generic queue. To set this flag, use the `/NOENABLE_GENERIC` qualifier with the `SET QUEUE`, `START/QUEUE`, or `INITIALIZE/QUEUE` command in the following format:

```
INITIALIZE/QUEUE/NOENABLE_GENERIC queue-name[:]
```

For more information about the `SET QUEUE`, `START/QUEUE`, or `INITIALIZE/QUEUE` command and qualifiers, see the *VMS DCL Dictionary*.

System Manager Release Notes

2.5 Batch and Print Queuing System

2.5.7 Queue Manager CPU Limit—Problem Corrected

V5.5-1 In VMS Version 5.5, if a queue was assigned a maximum CPU limit (with INITIALIZE/QUEUE/CPUMAXIMUM=time), that limit was ignored if either of the following conditions was true:

- If no CPU default was assigned to the queue (with INITIALIZE/QUEUE /CPUDEFAULT=time)
- If the job was submitted with an infinite CPU time (SUBMIT /CPUTIME=INFINITE)

In VMS Version 5.5-1, this problem has been corrected.

2.5.8 Queue Manager Stop Pending Status—Problem Corrected

V5.5-2 In VMS Version 5.4, if a system manager issued a STOP/QUEUE/NEXT command to stop a queue, and then issued a START/QUEUE command shortly thereafter, the stop_pending status set by the STOP/QUEUE/NEXT command was cleared. In VMS Version 5.5, the stop_pending status was not cleared by the START /QUEUE command.

In VMS Version 5.5-2, this problem has been corrected.

2.5.9 Queue Manager SYMDEL Message—Problem Corrected

V5.5-1 In VMS Version 5.5, in some cases when you shut down a node, and many queues were stopped within a very short timespan, the Queue Manager repeatedly signalled:

```
%QMAN-E-SYMDEL, unexpected symbiont process termination
-SYSTEM-S-NORMAL, normal successful completion
```

If the symbiont had been the processor for an autostart queue, these messages were also accompanied by a message such as:

```
%QMAN-I-QUEAUTOOFF, queue PRINTQ_01 is now autostart inactive
```

In VMS Version 5.5-1, this problem has been corrected. No SYMDEL messages appear when symbiont processes terminate normally, and autostart queues remain active for autostart when their symbiont processes terminate normally.

2.5.10 Saving Information in the Queue Database—Problem Corrected

V5.5-2 With VMS Version 5.5, when you backed up the queue database files, information about jobs was not saved. You normally do not need to save job information; however, if you backed up the disk containing the queue database files, and rebooted using the backup copy, previously existing jobs may have been missing from the queue database.

In VMS Version 5.5-2, this problem has been corrected. Job information is saved under the following conditions:

- When the queuing system has been shut down
- When the entire computer system has been shut down using the orderly shutdown procedure SYS\$SYSTEM:SHUTDOWN.COM

Job information is not saved when the queuing system is running. However, you can save information about queues, forms, and characteristics on a running queuing system by using the CONVERT/SHARE command as explained in the chapter on managing queues in the *Guide to Maintaining a VMS System*.

2.5.11 SUBMIT/USER and PRINT/USER Commands Provide Incorrect Account Name—Problem Corrected

V5.5-2 As of VMS Version 5.5, the SUBMIT/USER and PRINT/USER commands failed to provide the account name for the user specified with the /USER qualifier in the following locations:

- VMS Accounting Utility (ACCOUNTING) records
- SHOW QUEUE and SHOW ENTRY displays

Instead, the account name for the submitter was used. In VMS Version 5.5–2, this problem has been corrected.

2.5.12 SYNCHRONIZE Command—Problem Corrected

V5.5-1 In versions of VMS prior to Version 5.5, the SYNCHRONIZE command did not perform as documented. The *VMS DCL Dictionary* describes the behavior as follows:

If you specify the *job-name* parameter, the default queue is SYS\$BATCH.

However, the command incorrectly found the job with the specified job name on any batch queue.

In VMS Version 5.5–1, the SYNCHRONIZE command performs as documented.

In addition, if the job was submitted to a generic queue and that queue is specified, the SYNCHRONIZE command now searches for the specified job in the generic queue and in its target queues.

2.6 CLUSTER_CONFIG.COM Command Procedure—Updated to Read Undefined System Disks

V5.5-2 In VMS Version 5.5–2, the CLUSTER_CONFIG.COM command procedure has been updated to understand system disks that do not have a Logical Volume Name (LOGVOLNAM) defined. In this case, CLUSTER.CONFIG.COM now uses the translation of the logical name SYS\$SYSDEVICE as the name of the system disk.

2.7 Data Corruption on Disk with New Allocation Class

V5.5-2 In previous versions of VMS, if the allocation class of a controller was changed, or if the allocation of a node that was MSCP serving disks to the other nodes of a VAXcluster, the disk class drivers on the other nodes of the cluster reconnected to the controller after it rebooted with the new allocation class and resumed accessing its disks. This functionality could have caused disks within the cluster to be accessed using two different names, and disk corruption could result.

The allocation class is part of the unique identifier for devices being accessed through a particular controller. Each device within a cluster must have a unique name that is agreed upon by all the nodes of the cluster in order for access to that device to be synchronized across the cluster. The allocation class for a given controller should be verified as being equal to the allocation class in the local data structures on a reconnect to a previously known controller. This verification must occur before I/O is permitted on a new connection.

System Manager Release Notes

2.7 Data Corruption on Disk with New Allocation Class

In VMS Version 5.5-2, when the disk class driver establishes a new connection to a disk controller, it checks to see if this is a reconnect to a previously identified controller, or a connection to a previously unknown controller. If the connection being formed is to a controller that was previously identified, the allocation class of that controller is checked to make sure that it has not changed since the last connection to that controller was lost. If the allocation class has changed, the connection is disconnected and remains unable to reconnect, in order to prevent the corruption scenario described above. In order for the old node to form a connection to a controller that has been rebooted with a new allocation class, you must reboot the old node.

2.8 DECnet-VAX Notes

The release notes in this section pertain to DECnet-VAX software.

2.8.1 Adjusting NETACP Quotas with NETACP\$ Logical Names

V5.5-2

In VMS Version 5.5-2, you can define certain system logical names that control the resources allocated to the network ancillary control process (NETACP) in SYS\$MANAGER:SYLOGICALS.COM. By defining these system logical names, you can override the default resource allocation. These system logical names are:

- NETACP\$MAXIMUM_WORKING_SET (WSQUOTA)
- NETACP\$PAGE_FILE (PGFLQUOTA)
- NETACP\$EXTENT (WSEXTENT)
- NETACP\$ENQUEUE_LIMIT (ENQLM)
- NETACP\$BUFFER_LIMIT (BYTLM)

If these system logical names are not defined, default values for these resources will be provided by SYS\$MANAGER:LOADNET.COM when the NETACP process is created. The default values for these quotas will be adequate for most systems, and it is recommended that these system logical names remain undefined unless NETACP performance problems are observed.

The supported method of modifying these quotas is to use the logical names, not to modify the LOADNET.COM file. For example, if the NETACP process consistently exhibits an unacceptable page faulting rate, NETACP's WSQUOTA and WSEXTENT should be adjusted accordingly. The appropriate values for these quotas are configuration dependent, but in general, higher values are required for systems having very large node databases.

The following specific symptoms are indications that the NETACP process' BYTLM quota requires adjustment:

- "%SYSTEM-F-EXQUOTA, exceeded quota" is returned when attempting to turn on a line.
- A circuit will not transition into the "on" state but remains in the "on-synchronizing" state after service has been enabled; however, the circuit does start correctly once service is disabled.
- You can roughly estimate the BYTLM quota required for NETACP as follows:
 - a. Allow 3500 bytes for DECnet startup.
 - b. To this total, add the values resulting from the following multiplications (calculate the products for all lines):
receive_buffers * line_buffer_size

- c. Increase BYTLM by 7200 bytes for each circuit that has service enabled.

2.8.2 Cluster Synchronization Improved

V5.5-2 In very rare circumstances, some nodes in the cluster could not create logical links with the cluster alias. A correction has been made that improves synchronization for the cluster alias data structure.

Note

In a mixed cluster, older DECnet-VAX full routing nodes should install the same version of cluster alias logic introduced in the V5.5-2 NETACP and NETDRIVER. A DECnet-VAX upgrade kit SYS\$UPDATE:DECNETIV_0551A1.A should be copied to DECnet-VAX full routing nodes with VMS V5.5-1 or prior and installed using VMSINSTAL. A system reboot will be necessary after this installation is complete. (VMS V5.4 is the earliest version that is supported by the DECNETIV_0551A1.A upgrade kit.)

See the *VMS Version 5.5-2 Update Procedures* for detailed information on how to install the DECnet-VAX upgrade kit.

2.8.3 Event Logger—Problem Corrected

V5.5-2 In previous versions of VMS, the event logger (EVL) incorrectly displayed the message “Unknown counter type 822” instead of “Adjacency Down,” and “Unknown counter type 900” instead of “Peak Adjacencies” when all of the following conditions were present:

- The logging console was on
- Event 0.9 was enabled
- Circuits were set to 0

In VMS Version 5.5-2, this problem has been corrected. In addition, the text associated with the line and circuit counters in the EVL log has been updated to be consistent with the Network Control Program (NCP) line and circuit counters.

2.8.4 EXECUTOR NODE Corruption—Problem Corrected

V5.5-2 In VMS Version 5.5-2, a correction to the EXECUTOR NODE has been implemented. When you use NCP to set the EXECUTOR NODE state to shut, NETACP exits after all logical links are disconnected. Previously, nodes with their cluster alias set failed to transition to the off state. With the correction, even nodes with a cluster alias can transition to off when all of the logical links are terminated. If this EXECUTOR is also the routing node for the cluster, then all of the logical links using this cluster alias will also be disconnected when EXECUTOR NODE state is set to shut.

2.8.5 FDDI and Token Ring Device Support—Problem Corrected and Improvements Added

V5.5-2 In previous versions of VMS, the default line protocol and circuit type for Fiber Distributed Data Interface (FDDI) and Token Ring devices was incorrectly declared as Ethernet. In VMS Version 5.5-2, this problem has been corrected. The default line protocol is now declared as Token Ring.

System Manager Release Notes

2.8 DECnet-VAX Notes

In VMS Version 5.5-2, Fiber Distributed Data Interface (FDDI) or Token Ring lines fail to start if you define the protocol as Ethernet in the DECnet permanent database. If this occurs, you can start the line after you issue the Network Control Program (NCP) command PURGE LINE line-id PROTOCOL.

In VMS Version 5.5-2, FDDI-specific line parameters have been implemented (see Section 2.8.6).

2.8.6 FDDI Line Parameters for Phase IV Network Management

V5.5-2

In VMS Version 5.5-2, DECnet-VAX Phase IV network management support has been added for line parameters specific to the FDDI (Fiber Distributed Data Interface).

Although you can set NIF/SIF/ECHO target addresses and ECHO parameters in NCP, these parameters are presently used for informational purposes only. As of this release, this information is not passed to the device; therefore, you cannot use NCP to issue NIF/SIF/ECHO requests.

The following parameters have been added to the NCP command SET LINE:

- **ECHO DATA hex_byte**

Applies only to FDDI lines. ECHO LENGTH is the number of bytes of value. ECHO DATA is used to compose the next echo request frame that is sent to the address specified by ECHO TARGET. Hex_byte must be a string of exactly two hexadecimal digits. The default ECHO DATA is 55. ECHO DATA can be set in the volatile database, but it cannot be defined in the permanent database.
- **ECHO LENGTH count**

Applies only to FDDI lines. ECHO LENGTH is the number of bytes of type. ECHO DATA is used to compose the next echo request frame that is sent to the address specified by ECHO TARGET. Count must be a decimal value from 0 to 4478. The default ECHO LENGTH is 1. ECHO LENGTH can be set in the volatile database, but it cannot be defined in the permanent database.
- **ECHO TARGET address**

Applies only to FDDI lines. Specifies the address to which an echo request frame is sent. The default ECHO TARGET is 00-00-00-00-00-00. The ECHO TARGET can be set in the volatile database, but it cannot be defined in the permanent database.
- **NIF TARGET address**

Applies only to FDDI lines. Specifies the address to which the next NIF (Neighborhood Information Frame) request frame is sent. The default target is 00-00-00-00-00-00. NIF TARGET can be set in the volatile database, but it cannot be defined in the permanent database.
- **PROTOCOL FDDI**

Specifies the line protocol type to be FDDI.
- **REQUESTED TRT microseconds**

Applies only to FDDI lines. Specifies the requested value for the token rotation timer in microseconds. Microseconds must be a decimal integer in the range of 4000 to 167772. The default is 8000 microseconds.
- **RESTRICTED TOKEN TIMEOUT milliseconds**

System Manager Release Notes 2.8 DECnet-VAX Notes

Applies only to FDDI lines. Specifies the limit on how long a single restricted mode dialog may last before being terminated. Milliseconds must be a decimal integer in the range of 0 to 10000. The default is 1000 milliseconds.

- RING PURGER ENABLE option

- ON

Participate in the Ring Purger election and, if elected, perform the Ring Purger function. This is the default.

- OFF

Do not participate in the Ring Purger election. This parameter is to allow operation when certain nonconforming stations are on your ring; except for that case it should be left ON for improved ring reliability.

- SIF CONFIGURATION TARGET address

Applies only to FDDI lines. Specifies the address to which a SIF (Status Information Frame) configuration request frame is sent. The default configuration target is 00-00-00-00-00-00. SIF CONFIGURATION TARGET can be set in the volatile database, but it cannot be defined in the permanent database.

- SIF OPERATION TARGET address

Applies only to FDDI lines. Specifies the address to which a Status Information Frame (SIF) operation request frame is sent. The default operation target is 00-00-00-00-00-00. SIF OPERATION TARGET can be set in the volatile database, but it cannot be defined in the permanent database.

- VALID TRANSMISSION TIME microseconds

Applies only to FDDI lines. Specifies the maximum time between arrivals of a valid frame or unrestricted token. Microseconds must be a decimal integer in the range of 2500 to 5222. The default is 2621 microseconds.

The complete format for calling the SET LINE command is as follows:

```
SET LINE line-id parameter value
```

The following parameters have been added to the NCP command CLEAR LINE:

- ECHO DATA

Applies only to FDDI lines. Resets ECHO DATA parameter to its default value of 55 in the volatile database. Permanent database operations cannot be performed on this parameter.

- ECHO LENGTH

Applies only to FDDI lines. Resets the ECHO LENGTH parameter to its default value of 1 in the volatile database. Permanent database operations cannot be performed on this parameter.

- ECHO TARGET

Applies only to FDDI lines. Resets the ECHO TARGET parameter to its default value of 00-00-00-00-00-00 in the volatile database. Permanent database operations cannot be performed on this parameter.

- NIF TARGET

Applies only to FDDI lines. Resets the NIF TARGET parameter to its default value of 00-00-00-00-00-00 in the volatile database. Permanent database operations cannot be performed on this parameter.

System Manager Release Notes

2.8 DECnet-VAX Notes

- **REQUESTED TRT**
Applies only to FDDI lines. Resets the REQUESTED TRT parameter to its default value of 8000 microseconds in the volatile database.
- **RESTRICTED TOKEN TIMEOUT**
Applies only to FDDI lines. Resets the RESTRICTED TOKEN TIMEOUT parameter to its default value of 1000 milliseconds in the volatile database.
- **RING PURGER ENABLE**
Applies only to FDDI lines. Resets the RING PURGER ENABLE parameter to its default value of ON in the volatile database.
- **SIF CONFIGURATION TARGET**
Applies only to FDDI lines. Resets the SIF CONFIGURATION TARGET parameter to its default value of 00-00-00-00-00-00 in the volatile database. Permanent database operations cannot be performed on this parameter.
- **SIF OPERATION TARGET**
Applies only to FDDI lines. Resets the SIF OPERATION TARGET parameter to its default value of 00-00-00-00-00-00 in the volatile database. Permanent database operations cannot be performed on this parameter.
- **VALID TRANSMISSION TIME**
Applies only to FDDI lines. Resets the VALID TRANSMISSION TIME parameter to its default value of 2621 microseconds in the volatile database.

The complete format for calling the CLEAR LINE command is as follows:

```
CLEAR LINE line-id parameter value
```

2.8.6.1 SHOW LINE Display Modified

V5.5-2

The new SHOW LINE line-id CHAR display for FDDI lines in the Network Control Program (NCP) is as follows:

```
$ RUN SYSS$SYSTEM:NCP
NCP> SHOW LINE MFA-0 CHAR

Line Volatile Characteristics as of  5-JUN-1992 17:25:49
Line = MFA-0

Receive buffers           = 10
Controller                = normal
Protocol                  = FDDI
Service timer             = 4000
Hardware address          = 08-00-2B-1C-12-16
Device buffer size        = 1498
Requested TRT             = 0          <- FDDI-specific line
Valid transmission time    = 2621      chars begin here.
Restricted token timeout   = 0
Ring purger enable        = off
NIF target                = 00-00-00-00-00-00 <- From here
SIF configuration target   = 00-00-00-00-00-00 down, these
SIF operation target      = 00-00-00-00-00-00 parameters
Echo target               = 00-00-00-00-00-00 allow only
Echo data                 = 00          volatile
Echo length               = 0          database
                           operations
```

System Manager Release Notes 2.8 DECnet-VAX Notes

Also, the format of the SHOW LINE line-id STATUS display has been modified for all lines. The format of this display is now similar to the format of the line characteristic display. This display was modified to include additional FDDI-specific line parameters such as "Negotiated TRT."

The following is an example of a display that includes FDDI-specific parameters:

```
NCP> SHOW LINE MFA-0 STAT
Line Volatile Status as of 5-JUN-1992 17:30:33

Line = MFA-0
State = on
Negotiated TRT = 99840 <- FDDI-specific
Duplicate address flag = unknown status parameters
Upstream neighbor = 08-00-2B-1C-0D-BB start here
Old upstream neighbor = 00-00-00-00-00-00
Upstream neighbor DA flag = unknown
Downstream neighbor = 00-00-00-00-00-00
Old downstream neighbor = 00-00-00-00-00-00
Ring purger state = off
Ring error reason = no error
Neighbor PHY type = A
Link error estimate = 15
Reject reason = none
```

2.8.7 NCP Command Prevents Unauthorized Connections to the Mail Server

V5.5-2

MAIL now enables system privileges when it opens a DECnet connection to a remote mail server. The system manager can restrict outgoing access on the mail server object by using the following NCP command:

```
$ RUN SYSS$SYSTEM:NCP
NCP> DEFINE OBJECT MAIL OUTGOING CONNECT PRIVILEGE SYSPRV
```

This command prevents unauthorized users from placing connections on the mail server object.

2.8.8 NCP Commands for DNS Interface

V5.5-2

In VMS Version 5.5-2, commands have been added to the NCP (Network Control Program) to allow DECnet-VAX Phase IV nodes to use the DNS (Distributed Name Service) interface. The DNS interface allows the executor to search for node information in a DNS namespace if that node information is not already present in the volatile node database.

Do not enable the DNS interface on a Phase IV node unless the node resides in a network that has begun migration to DECnet/OSI. If a Phase IV node has access to a DNS namespace that is populated with the node information, you can use this interface in lieu of maintaining a permanent node database.

The following are the new commands for NCP:

- **SET/DEFINE EXECUTOR DNS INTERFACE**

Specifies whether the local node will use DNS to update the node information in the volatile database. There are two options for EXECUTOR DNS INTERFACE:

- **ENABLED**

System Manager Release Notes

2.8 DECnet-VAX Notes

Enable this interface if you have begun migration to a DECnet/OSI network and a DNS namespace is available to the executor node. ENABLED specifies that the first time a node not present in the volatile node database is referenced, DNS searches for the node information. This node information is retained in the volatile node database for future use.

– DISABLED

Specifies that the local node will not use DNS to search for node information. The DNS interface is disabled by default.

- SET EXECUTOR DNS namespace
Specifies the DNS namespace used by the DNS interface. DNS namespace is a string of 1 to 256 alphanumeric characters that can include dollar signs, hyphens, and underscores.
- SET EXECUTOR IDP
Specifies the IDP of the ISO address. This is a string of 1 to 22 hexadecimal digits.
- CLEAR EXECUTOR DNS INTERFACE
Removes the EXECUTOR DNS INTERFACE parameter from the volatile database.
- CLEAR EXECUTOR DNS namespace
Removes the EXECUTOR DNS namespace parameter from the volatile database.
- CLEAR EXECUTOR IDP
Removes the EXECUTOR IDP parameter from the volatile database.

2.8.9 NETACP Allocation of Node Counter Blocks Changed

V5.5-2

The allocation of node counter blocks has changed. Previously, NETACP allocated a fixed number of node counter blocks at startup. The number allocated (512) determined the maximum number of nodes that could concurrently have a logical link open with a single DECnet node.

When a logical link could not be completed because all node counter blocks were in use, additional DECnet connections would be rejected with the reason SS\$_CONNECFAIL. Node counter blocks are allocated on a per-node basis and this could have happened with any system attempting to establish logical links with more than 512 nodes at once.

This limitation had the greatest effect on large PATHWORKS servers. A PATHWORKS client typically will have only one link per node, but there will be many client nodes for a single server node. In VMS Version 5.5-2, this limitation is now removed.

Node counter blocks are now allocated in groups of 32. This reduces the initial size of NETACP's initial running image, but allows it to increase as needed. This will benefit small configurations that do not maintain concurrent logical links to large numbers of different DECnet nodes.

This change may cause DECnet event 3.2 (Database Reused) to be seen. The event is informational and may have been seen before this change. It will now be possible for this message to be seen in smaller networks, or sooner in larger ones.

System Manager Release Notes 2.8 DECnet-VAX Notes

DECnet event 3.2 may also be seen frequently on VMS systems running DECmcc. This can result from how rules are defined and the resulting polling activity. DECmcc tends to have many short-lived connections to many different nodes. This can cause relatively rapid recycling of node counter blocks.

If these events should be frequent enough to be annoying or cause operator logs to grow excessively, it is recommended they be turned off with these commands:

```
$ NCP CLEAR KNOWN LOGGING EVENTS 3.2      !In Volatile Data Base
$ NCP PURGE KNOWN LOGGING EVENTS 3.2      !In Permanent Data Base
```

This event is generated when a connection is created with a node that has no node counter block associated with it, there are no unused node counter blocks available, and there is at least one node that has a node counter block that is inactive (no current logical link with that node). Reusing inactive node counter blocks binds the number that must be accommodated in NETACP's address space to the maximum number of nodes concurrently (since NETACP startup) connected plus 31.

Idle node counter blocks are kept in a first-in/first-out (FIFO) queue so their information is retained as long as possible before reuse. If another connection is created for a node that has an inactive node counter block, it is removed from the inactive queue and remains associated with that node. This helps retain information about the nodes communicated with most frequently for the longest period of time, and acts as a node counter block cache for these nodes.

In conjunction with this change, the hash table used to look up node counter blocks and the hashing algorithm has been changed to operate faster and with less CPU overhead.

2.8.10 NETDRIVER PATH SPLIT POLICY Call—Problem Corrected

V5.5-2 In previous versions of VMS, a problem existed in NETDRIVER such that an INVEXCEPTN crash could result when the PATH SPLIT POLICY was set to INTERIM. In VMS Version 5.5-2, this problem has been corrected.

2.8.11 NETACP Endnode Failover—Problem Corrected

V5.5-2 Endnode failover has been corrected in NETACP. In previous versions of VMS, failover on dual circuits failed to keep the logical link alive because the failed circuit continued to be the path selected. In VMS Version 5.5-2, the alternate circuit is selected and the logical link continues to function.

2.8.12 NDDRIVER Queue Corruption—Problem Corrected

V5.5-2 In previous versions of VMS, a condition existed that caused a queue corruption in NDDRIVER when downline loading multiple targets from SMP hosts. A SSRVEXCEPT in NDDRIVER could result when multiple MOM processes were executing. In VMS Version 5.5-2, this problem has been corrected.

2.8.13 NCP Module Configurator—Problem Corrected and Improvements Added

V5.5-2 In VMS Version 5.5-2, the following improvements have been made to the NCP module configurator:

- A problem has been corrected that, under certain circumstances, caused the NICONFIG image to exit prematurely. This premature exit of the NICONFIG image caused the loss of the context of the module configurator's volatile circuit database and the return of inconsistent error messages in response to the next module configurator command issued.

System Manager Release Notes

2.8 DECnet-VAX Notes

- Support has been added for the CLEAR MODULE CONFIGURATOR KNOWN CIRCUITS ALL command.
- Device tables have been updated so that the SHOW MODULE CONFIGURATOR KNOWN CIRCUITS STATUS command displays device type names instead of device type numbers. This is true only for the home area.

2.8.14 NCP SHOW KNOWN NODES Display—Problem Corrected

V5.5-2 In previous versions of VMS, a problem existed in the Network Control Program (NCP) when you entered the SHOW KNOWN NODES command. If the last node of the home area was reachable and this last node had no name associated with it in the volatile database, the SHOW KNOWN NODES display omitted the first node of the area immediately following the home area.

In VMS Version 5.5-2, this problem has been corrected.

2.8.15 Patched Images for DNS Version 1.1

V5.5-2 This update contains patched images for DNS Version 1.1. If you have installed DECnet/OSI for VMS V5.5 of DECnet/VAX on your system, be aware that DECnet/OSI for VMS V5.5 contains DNS Version 2.0. The DNS patches in this update were also included in DNS Version 2.0. If DNS Version 2.0 is present on the system, the updated images for DNS Version 1.1 will not be installed.

2.8.16 Phase IV Support for DSW-21 and DSW-41/42 Devices

V5.5-2 VMS Version 5.5-2 adds DECnet-VAX Phase IV network management support for the DSW-21 single-line serial communications device and the DSW-41/42 single- or dual-line serial communications device.

The default protocol for these devices is DDCMP POINT. The associated driver, ZTDRIIVER, is included in VAX WAN Device Drivers Version 1.2. The Network Control Program (NCP) network management mnemonic for these devices is DSW, and for service purposes the mnemonics for these devices are DSW for the DSW-21 and DW4 for the DSW-41/42.

2.9 DECwindows

The release notes in this section pertain to the VMS DECwindows software supplied with the VMS operating system.

2.9.1 Dashed Lines Drawn Incorrectly—Problem Corrected

V5.5-1 In the VMS Version 5.5 X11 server for VS4000 base system graphics, there was a problem with wide on and off dashed lines and segments that were drawn using the following logical functions:

- GXandReverse
- GXxor
- GXnor
- GXequiv
- GXinvert
- GXorReverse
- GXnand

The visual effect was that the lines were drawn randomly. There may also have been math errors reported in the server error log file.

In VMS Version 5.5-1, the errors in the logical functions have been corrected.

2.9.2 DECwindows Monitor Density Is Now Definable

V5.5-2

It is now possible to define the DECwindows server's monitor density as a value separate from the server density. In the past, it was possible only to define the value `DECW$SERVER_DENSITY` in the file `SYS$MANAGER:DECW$PRIVATE_SERVER_STARTUP.COM`. This value is used to determine the font size, either 75 dots per inch or 100 dots per inch. As such, this value is limited to either of these two values.

In addition to being used to determine the font set, this value was used in calculating the physical width of the screen, which is available from several X Library routines. Since few monitors are actually 75 or 100 dots per inch, it was impossible to display physical length items on the screen.

By setting `DECW$MONITOR_DENSITY` to the actual value in the file `SYS$MANAGER:DECW$PRIVATE_SERVER_SETUP.COM`, it is possible to obtain correct values for the width and height of the screen using X Library routines.

To calculate the actual monitor density, measure across the visible portion of the screen (in inches) and divide by the pixel width of the screen. You can obtain the pixel width of the screen by looking at the value of the logical name `DECW$XSIZE_IN_PIXELS`, which is defined in the logical name table `DECW$SERVER0_TABLE`. Generally, this will be either 1024 or 1280, depending on the graphics adapter on your system.

For example, if you have a VRT19 monitor and SPX graphics, you would make this calculation:

```
1280 pixels / 13.5 inches = 94.81 dpi
```

Rounded to the nearest integer, this becomes a 95 dpi monitor, and the correct entry to put into the private server setup file is:

```
$ DECW$MONITOR_DENSITY == "95"
```

If you have a multihead workstation, this can be set on a per-monitor basis; for example:

```
$ DECW$MONITOR_DENSITY == "95,75"
```

Setting the monitor density far from the server density can cause problems with "what-you-see-is-what-you-get" (WYSIWYG) applications, such as DECwrite, since it is not currently possible to scale the 75 and 100 dpi fonts to match the actual monitor density.

To set the font size for the server so that it will most closely match the density of the color screen of the workstation in the previous example, you might use the following command in conjunction with that example:

```
$ DECW$SERVER_DENSITY == "100"
```

The default value for the monitor density is the server density. The following definition will set the font size to 100 dpi (since this is the value provided for the primary head of the workstation), and the two monitor densities to 100 and 75, respectively:

```
$ DECW$SERVER_DENSITY == "100,75"
```

If the value of the server density for the primary screen is neither 75 nor 100 dpi, the 75 dpi fonts are supplied as a default.

System Manager Release Notes

2.9 DECwindows

2.9.3 DECwindows Server Incompatibility with DEC PHIGS

V5.5-2 VMS Version 5.5-2 includes DECwindows server files that are not compatible with DEC PHIGS Versions 2.3A and 2.3B. DEC PHIGS is used by 3D applications. To provide your users with a compatible version of DEC PHIGS you must install DEC PHIGS Version 2.3C following the successful completion of this update.

If you cannot install DEC PHIGS Version 2.3C after this update, then you may choose from the following options to determine your system environment:

- Continue and complete the entire update, including the DECwindows files. All bug fixes and new hardware support included in this kit will be available to users. Note that 3D applications do not work until you install DEC PHIGS Version 2.3C.
- Continue and complete this update with the exception of the DECwindows files included in this kit. 3D applications will continue working, however DECwindows bug fixes and new hardware graphics support provided by this kit will not be available to users. Note that this option requires you to reapply the VMS Version 5.5-2 update after you install DEC PHIGS Version 2.3C, to provide the DECwindows bug fixes and new hardware graphics support.
- Exit the update at this time and apply the VMS Version 5.5-2 update after you install DEC PHIGS Version 2.3C.

2.9.4 DECwindows Use of SMP with DECsound and SODRIVER

V5.5-2 When you run VMS Version 5.5-2 with full SMP checking turned on (with the system parameter MULTIPROCESSING set to 2), DECsound is not supported. This is because of an error in the AM79C30A audio chip driver. When multiprocessing is enabled, even if you are on a uniprocessor machine, the audio chip driver fails to load, and DECsound does not function properly.

If you install DECwindows Motif Version 1.1, do not keep full SMP checking enabled and use the DECsound editor at the same time. DECwindows Motif Version 1.1 has its own version of the AM79C30A audio chip driver, named SODRIVER.EXE. This driver causes a system crash with an SPLIPLHIGH failure.

Note

This information applies to DECwindows Motif Version 1.1. DECsound is not the only application that uses SODRIVER; there is also audio support in the CDA Viewer, and therefore audio applications such as the CDA Viewer are also unsupported with full SMP checking turned on.

2.9.5 SoftPC with DECwindows Server—Keyboard Problem

V5.5-2 If you have the SoftPC layered product installed on your system while using the graphics console head, and the DECwindows server is shut down, the keyboard echoes duplicate characters back to the console. You can work around this problem if you completely release every key prior to pressing the next key. To hold down the shift key, delete the duplicate character to get only one of those characters. This problem does not occur when you use the alternate console head.

2.9.6 Tiled Operations Drawn in Wrong Window—Problem Corrected

V5.5-1 In the VMS Version 5.5 X11 server for VS4000 base system graphics, a problem in certain tiled operations caused objects to be drawn in the wrong window. The visual effect was that an object to be drawn in one window was instead drawn in a window that was on top of the intended window.

In VMS Version 5.5-1, this problem has been corrected.

2.10 Delta/XDelta Utility (DELTA/XDELTA)

The release notes in this section pertain to the Delta/XDelta Utility (DELTA/XDELTA).

2.10.1 Delta/XDelta Utility—Booting New VAX Computers

V5.5-2 Table 2-1 lists the commands used to boot the VMS Version 5.5-2 operating system with DELTA/XDELTA on recently released VAX computers.

Table 2-1 VAX Boot Commands

Boot Command	VAX Computer
BOOT/R5:n devname ¹	MicroVAX 3100-90 VAXstation 4000-90
BOOT/n devname ¹	VAX 4000-100 VAX 4000-400 VAX 4000-600

¹Where *n* is a value from Table 2-2 and *devname* is the name of the device (in the form *ddcu*) from which you are booting the system.

Table 2-2 lists the command qualifier values used to boot the VMS Version 5.5-2 operating system with DELTA/XDELTA on recently released VAX computers.

Table 2-2 VAX Boot Command Qualifier Values

Value	Description
0	Normal, nonstop boot (default)
1	Stop in SYSBOOT
2	Include XDELTA with the system, but do not take the initial breakpoint
6	Include XDELTA with the system, and take the initial breakpoint
7	Include XDELTA with the system, stop in SYSBOOT, and take the initial breakpoint at system initialization

Note

When you deposit a boot command qualifier value in R5, make sure you include any other values you would normally deposit. For example, if you were depositing the number of the system root directory from which you were booting *and* an XDELTA value, R5 would contain both values.

See the *VMS Delta/XDelta Utility Manual* for more information about using the Delta/XDelta Utility on VAX computers.

System Manager Release Notes

2.11 Disk Class Drivers

2.11 Disk Class Drivers

This section describes corrections to disk class drivers.

2.11.1 MSCPCLASS Error—Problem Corrected

V5.5-1 A device configured with a path to both a served controller and a hierarchical storage controller (HSC) could have, upon failover (from the served controller to the HSC port), occasionally caused an improper MSCPCLASS bugcheck to occur. An additional check has been made to prevent this situation.

2.11.2 Opcode for MSCP Message Improperly Set—Problem Corrected

V5.5-1 Under certain error recovery conditions, the opcode for a mass storage control protocol (MSCP) message would not be set properly and would cause unpredictable results. The opcode field is now set properly.

2.11.3 Shutdown Error While Copy Operation in Progress—Problem Corrected

V5.5-1 During system shutdown, if a phase I shadow set member volume control block (VCB) was deallocated while a copy operation was in progress, the driver might have attempted to access fields in the now deallocated structure. Doing so could have resulted in a system failure with an INVEXCEPTN error. An additional check has been added to prevent the system failure while a shutdown is pending.

2.11.4 Volume Shadow Set Members Race Condition—Problem Corrected

V5.5-1 While a phase I volume shadow set member was experiencing a high rate of recoverable errors, a race condition could occur between an operator-requested DISMOUNT command and the spontaneous removal of the member by the class driver. Further checks have been added to prevent this condition.

2.12 Distributed Lock Manager

This section describes the corrections to the VMS distributed lock manager.

2.12.1 Block Transfer to an Unavailable Node Caused System Failure—Problem Corrected

V5.5-1 A block transfer to a node whose VMS\$VAXcluster connection breaks caused a CNXMGRERR bugcheck at CLUSTRLOA+3766. If the connection between two nodes was disrupted in the middle of a cluster server process (CSP) block transfer, and the block transfer completed while the connection was broken, the last message sent had a response value of zero. The zero RSPID response now causes the block transfer to be redone.

2.12.2 Lock Range Value Corruption—Problem Corrected

V5.5-1 Noncontiguous fields caused a lock range value corruption that caused a LOCKMGRERR machine check at CLUSTRLOA+8DF8. This problem has been corrected.

2.12.3 Remastering Routine Race Condition—Problem Corrected

V5.5-1 A race condition occurring in the remastering routine caused a LOCKMGRERR machine check at CLUSTRLOA+5F24.

If a new master failed during a remaster operation, and a rebuild message was sent before the new master received a shutdown message, the system would fail. The rebuild messages no longer cause a system failure.

2.12.4 RSPFATAL Status Caused System Failure—Problem Corrected

V5.5-1 A new status called RSPFATAL was introduced in VMS Version V5.5. The status is sent to a remote node that has sent invalid data. The result is a system crash on both of the nodes involved in a locking operation.

The status field was not being filled in. When the field was checked later, if the old data in the status byte happened to equal the value of RSPFATAL, the system failed.

With VMS Version 5.5-1, the status field is set to have the correct status.

2.12.5 System Parameter Added to Lock Manager

V5.5-2 VMS Version 5.5-2 adds a system parameter named PE1 to limit the size of a lock tree that you can move to another master. When PE1 is set to a non-zero value, the lock manager only considers trees for movement that have fewer locks than the value of the parameter. When set to 0 (the default value), the lock manager considers any size tree for movement. The parameter is node specific in that it only limits a specific node's ability to move a tree to another node. Thus, you can set PE1 to a different value on each node.

PE1 is a dynamic parameter and thus you can modify it without a system reboot. Digital recommends that you use the default value of 0.

2.13 InfoServer Client—Startup Behavior Change

V5.5-2 In previous versions of VMS, the documentation stated that to start the InfoServer Client for VMS the following command should be included in SYS\$MANAGER:SYSTARTUP_V5.COM:

```
$ @SYS$STARTUP:ESS$STARTUP CLIENT
```

In VMS Version 5.5-2, the p1 argument CLIENT is checked for its presence and spelling to load the disk client driver ESS\$DADDRIVER.EXE. In previous versions of VMS, any non-blank P1 argument caused the disk client driver to be loaded.

2.14 LAT Not Supported on DEQNA

V5.5-1 Beginning with VMS Version 5.5-1, the DEQNA network device is supported by fewer network protocols. The LAT driver for the LAT protocol now uses a new interface with the Ethernet drivers. Because this new interface is not supported on the DEQNA, you cannot use the LAT protocol with that Ethernet device. If your system has a DEQNA device, contact your Digital Services representative to make arrangements to have your DEQNA device upgraded to a DELQA.

2.15 Mount Utility (MOUNT)

This section describes corrections and one suggested workaround for the VMS Mount Utility.

2.15.1 Bound Volume Set Problem Using Volume Shadowing Phase I and Phase II—Suggested Workaround

V5.5-1 A mount request for a shadowed bound volume set may result in a fatal error if more than 5 units are specified. For example:

```
$ MOUNT/SYSTEM/BIND=PROD_B -  
DSA4100:/SHADOW=($1$DUA12), DSA4101:/SHADOW=($1$DUA14), -  
DSA4102:/SHADOW=($1$DUA16), DSA4103:/SHADOW=($1$DUA13), -
```

System Manager Release Notes

2.15 Mount Utility (MOUNT)

```
DSA4104:/SHADOW=($1$DUA20),DSA4105:/SHADOW=($1$DUA28) -  
BR0, BR1, BR2, BR3, BR4, BR5
```

```
%MOUNT-F-MAXDEV, too many devices
```

When a multiple-member virtual unit configuration is required, a workaround can be implemented. For example, to initially create the bound volume set, enter a command similar to the following:

```
$ MOUNT/BIND=DATADISK DSA1001:/SHADOW=$1$DUA1: DATA1
```

This device becomes the root volume of the volume set. Additional members of the volume set may then be created and mounted as follows:

```
$ MOUNT/BIND=DATADISK DSA1002:/SHADOW=$1$DUA2: DATA2  
$ MOUNT/BIND=DATADISK DSA1003:/SHADOW=$1$DUA3: DATA3  
$ MOUNT/BIND=DATADISK DSA1004:/SHADOW=$1$DUA4: DATA4
```

2.15.2 MOUNT Commands Delayed Other Nodes—Problems Corrected

V5.5-1

If you used the following command, MOUNT incorrectly computed the number of devices being mounted and then compared it against the number of labels specified:

```
$ MOUNT/SYSTEM DSA1313:/SHADOW=$254$DUA92: label
```

When the number of devices did not match the number of labels specified, MOUNT issued an error that resulted in a device lock being left in protected write (PW) mode. As a result, the next MOUNT/CLUSTER command for the same shadow set would hang the cluster server process (CSP) waiting for the device lock to be dequeued.

Also, when you used a MOUNT command and, as in the following example, specified only one of the shadow set members, MOUNT tried to add the second member automatically:

```
$ MOUNT/CLUSTER DSA1313:/SHADOW=$254$DUA92: label
```

Because of a software error, the lock to the device being automatically included was left in PW mode. As a result, the next MOUNT/CLUSTER command for the same shadow set would hang the CSP waiting for the device lock to be dequeued.

This problem has been corrected.

2.15.3 MOUNT Command Caused CPUs to Halt—Problem Corrected

V5.5-1

When mounting a multimember phase II volume shadow set in a CI- or NI-based cluster, all CPUs in the cluster were halted except for the CPU that issued the command. In VMS Version 5.5-1, CPUs are not halted.

2.15.4 Shadow Sets Improperly Allowed—Problem Corrected

V5.5-1

Physical units in phase II volume shadow sets could be made members of two different shadow sets. Physical units can now be part of only one shadow set.

2.15.5 Shadow Set Failure with Two-Member Volume Shadow Set—Problem Corrected

V5.5-1

Mounting a two-member phase II volume shadow set by specifying only one of the disks failed if you specified the second member first. For example, if a shadow set consisted of DEV1 and DEV2, then the following command failed to mount the shadow set:

System Manager Release Notes 2.15 Mount Utility (MOUNT)

\$ MOUNT/SYSTEM DSA1/SHAD=DEV2 LABEL

%MOUNT-F-DEVCOUNT, number of devices must match number of volumes

This problem has been corrected.

2.15.6 Shadow Set Logical Names Were Improperly Defined—Problem Corrected

V5.5-1 Logical names were not defined correctly for shadow sets. The logical names *DISK\$label* and *LOGICAL_NAME* (if given on the command line) both pointed to the last physical device in the shadow set. The logical names are now defined correctly.

2.15.7 Shadow Set Members Failed—Problem Corrected

V5.5-1 Binding shadow sets into a volume set did not work correctly when members were automatically included.

This problem has been corrected.

2.15.8 Tape Compaction Problems for TA90E/91—Problem Corrected

V5.5-1 The Mount Utility changed the expected data compaction behavior of TA90E/91 tape devices so that the compaction operation would not be enabled properly. The Mount Utility has been fixed to properly enable the tape drive data compaction operation.

2.15.9 Tape Density Improperly Set—Problem Corrected

V5.5-1 The MOUNT/FOREIGN command would incorrectly set the magnetic tape density to the low density. Magnetic tape density for multiple density drives is now set properly.

2.15.10 TUDRIVER During Mount Operation—Problem Corrected

V5.5-1 In previous versions of VMS, when you mounted tapes controlled by TUDRIVER, a system failure occurred. In VMS Version 5.5-1, this problem has been corrected.

2.16 New Operating System License Types

V5.5-2 If you purchased a new VAX computer with your VMS Version 5.5-2 operating system software, you may have received new, more flexible VMS licenses now offered with many VAX processors.

Note

Existing license types, including the VAX VMS license types described in the *VMS Version 5.5 Upgrade and Installation Manual*, are still supported. VAX processors that use them can coexist in a cluster with VAX processors that use the new license types.

These new licenses are as follows:

Operating System Base License

This license grants the right to non-interactive use of the remote batch, print, application and computing services of the VMS operating system on a single processor, and authorizes one direct login (for system management purposes only). This license is a prerequisite for the Interactive User Licenses (described next).

System Manager Release Notes

2.16 New Operating System License Types

Note that to provide greater flexibility in the selection of databases, the new licenses do not include the license rights for Rdb (which was included with previous VAX VMS licensing). The Rdb/VMS Run-Time license is available separately and as part of NAS 300 and NAS 400 integrated software products.

The Product Authorization Key (PAK) for the VMS Operating System Base License includes the following information:

- Product Name = BASE-VMS-250136
- Units = Processor-specific quantity
- Activity = A

Interactive User License

This license grants the right to interactive use of the VMS operating system, provided you have previously installed the appropriate Operating System Base License on your VAX computer.

The Product Authorization Keys (PAKs) for the Interactive User License include the following information:

- For a specific number of users:
 - Product Name = VMS-USER
 - Units = Number of users * 100
 - Activity = CONSTANT=100
- For an unlimited number of users:
 - Product Name = VMS-USER
 - Units = Processor-specific quantity
 - Activity = A

Interactive user licenses have a NO_SHARE attribute and remain with the initial host computer. You can add interactive users to the computer at any time, by specifying the same node name on the additional Interactive User License PAK and by following the license combination procedure described in the *VMS License Management Utility Manual*.

2.17 Operator Communication Manager (OPCOM) Restart

V5.5-2

In previous version of VMS, if the Operator Communication Manager (OPCOM) process failed, you had to restart it manually.

With VMS Version 5.5-2, if OPCOM fails, it automatically attempts to restart itself. OPCOM also leaves a process dump file named SYS\$SYSTEM:OPCOM.DMP. This file allows Digital representatives to determine the cause of the failure. If you encounter a software problem causing the OPCOM process to fail, include this file when you submit a software problem report (SPR).

OPCOM should normally be able to restart itself automatically. However, if OPCOM stops running and does not restart, you can manually restart the process, as you did prior to VMS Version 5.5-2, by entering the following command from the system manager's account (SYSTEM):

```
$ @SYS$SYSTEM:STARTUP OPCOM
```

For more detailed information on OPCOM, see the *Guide to Maintaining a VMS System*.

2.18 PATHWORKS PC Clients Can Interfere with InfoServer Clients

2.18 PATHWORKS PC Clients Can Interfere with InfoServer Clients

V5.5-2 Some PATHWORKS PC clients have faulty Ethernet adapters. These clients send VMS servers network messages that interfere with InfoServer clients.

To protect InfoServer clients, the next major release of VMS will filter messages from PATHWORKS PC clients. As a result, PC clients with faulty adapters might be unable to connect to VMS servers.

If you power on your PC and do not connect automatically, the PC might have a faulty Ethernet adapter. *Do not power cycle your PC.* To determine whether the adapter is faulty, start the client software by typing the normal command. If you are still unable to connect to a server, replace the PC Ethernet adapter.

2.19 POSIX Version 1.0 Users Should Upgrade to POSIX Version 1.1

V5.5-2 Digital strongly recommends that VMS POSIX Version 1.0 users upgrade to the VMS POSIX Version 1.1 kit before or after applying the VMS Version 5.5-2 update. Failure to do so may result in system failure and data corruption.

2.20 RMS Journaling

The release notes in this section pertain to the RMS Journaling Services.

2.20.1 Invalid Journals—Workaround

V5.5-2 In VMS Version 5.5-2, if a journal is corrupted, or if BACKUP operations have replaced the journal with another file that is not a journal, the commands SET FILE/NOAI_JOURNAL and SET FILE/NOBI_JOURNAL may fail with the following message:

```
%SET-F-IVJFILE, invalid journal file 'file_spec'
```

To correct this, delete the invalid journal file. The original SET FILE command then succeeds. If the file that was substituted for the journal is valuable, remember to make a copy of that file before you delete the invalid journal.

2.20.2 RMS Journaling with DCL Command RECOVER—Problem Corrected

V5.5-2 In VMS Version 5.5-2, the /UNTIL=TIME qualifier of the RECOVER command has been changed from a positional qualifier to a command qualifier. This means that a single RECOVER command can specify only one time, instead of multiple times as before.

Multiple times may still be specified on separate lines; for example:

```
$ RECOVER/RMS_FILE /forward /until=20-feb-1992:10 file_1  
$ RECOVER/RMS_FILE /forward /until=21-feb-1992:11 file_2
```

This change corrects a problem of combining the /UNTIL=TIME qualifier with wildcards, and is consistent with the warning that the use of different times can lead to inconsistent files.

System Manager Release Notes

2.21 SET PREFERRED PATH Problem—Suggested Workaround

2.21 SET PREFERRED PATH Problem—Suggested Workaround

V5.5-2 Failover may not occur as expected for VAX CI nodes that serve dual-pathed HSC disks to satellites when you have specified a preferred path using the \$QIO function IO\$_SETPRFPATH.

The preferred path feature is designed to have the local VAX CI node and satellite nodes (served through the MSCP server) use a specific path as their first attempt to locate and mount a disk. If the preferred path fails, the expected behavior is to fail over to an alternate path.

Mount attempts from the local VAX CI node will successfully fail over to the alternate path. However, satellite node mount attempts may fail because the MSCP server always tries to access the disk using the original preferred path; the server does not fail over to the alternate path. This behavior impacts only satellite mount attempts. This restriction will be fixed in a future release.

To work around this problem, clear the preferred path setting on the VAX CI node by specifying the local node name (instead of the node name of the HSC controller) as the preferred path.

See the *VMS I/O User's Reference Manual: Part I* for more information about setting a preferred path.

2.22 System Management Utility (SYSMAN) Server Hang

V5.5-1 The System Management Utility (SYSMAN) could hang in some instances, particularly when certain operations were interrupted with Ctrl/C or Ctrl/Y. A SYSMAN hang can cause the CLUSTER_SERVER process to stop running, which can cause a system bottleneck. The SYSMAN logic has been updated to improve the interaction between the SMISERVER process and the CLUSTER_SERVER process, which eliminates a possible system bottleneck.

2.23 System Parameter LGI_BRK_TERM—Problem Corrected

V5.5-2 In previous versions of VMS, intrusion detection records generated as a result of login failures or break-in attempts, which occurred at the DECwindows Login or Pause Session dialog boxes, did not properly obey the setting of the System Generation Utility (SYSGEN) system parameter LGI_BRK_TERM. As a result, the intrusion records always included the _WSAn: input device.

In VMS Version 5.5-2, this problem has been corrected. If you set the LGI_BRK_TERM system parameter to 0, the resulting intrusion record contains only the failing user name.

2.24 System Symbol Definitions for High-Level Languages

V5.5-2 In VMS Version 5.5-2, VMS provides new modules for the file SYS\$LIBRARY:STARLETSD.TLB, from which some high-level language compiler products (such as BASIC, FORTRAN, and Pascal) build language-specific equivalents of the VMS system symbol definition library STARLET.MLB. In previous versions of VMS, some sets of definitions were inadvertently omitted from STARLETSD.TLB, making them unavailable to high-level language programmers.

System Manager Release Notes

2.24 System Symbol Definitions for High-Level Languages

In VMS Version 5.5–2, the following modules have been added to STARLETS.D.TLB:

\$CLIMSGDEF	- Callable CLI\$ routine return statuses
\$CONVDEF	- Callable CONV\$ routine structure definitions
\$CONVMGDEF	- Callable CONV\$ routine return statuses
\$DDTMSGDEF	- Distributed Transaction Manager return statuses
\$DISMMSGDEF	- \$DISMOUNT system service return statuses
\$FDLMSGDEF	- Callable FDL\$ routine return statuses
\$LICENSEDEF	- License facility messages
\$MAILMSGDEF	- Callable MAIL\$ routine return statuses
\$MOUNDEF	- \$MOUNT system service return statuses
\$VAXDEF	- VAX hardware model codes

To make these new definitions available, you must reinstall the language compiler kit after you apply the VMS Version 5.5–2 update. Note that some languages (such as VAX Ada and VAX C) include VMS system symbol definition files on their compiler kits. For these languages, the new definitions may not be available until the compiler kits are updated. See the appropriate installation guides for the respective language compiler products for more information.

2.25 VAX 6000–600 Warm Start for Systems with Battery Backup—Problem Corrected

V5.5-1 If your VAX 6000-600 system contains a battery backup unit and recovered successfully from a power failure, the system halted at the next occurrence of any system error, even if the error was potentially recoverable. This halt prevented nonrecoverable hardware-related errors from causing bugcheck crashes.

In VMS Version 5.5–1, this problem has been corrected.

2.26 VAX 62xx and 63xx Systems with KLESI or UNIBUS Adapters Enabled

V5.5-1 In VMS Version 5.5, the primary CPU backup cache on VAX 62xx and 63xx systems with KLESI or UNIBUS adapters was disabled. Having this cache disabled resulted in a 50 percent reduction in performance. In VMS Version 5.5–1, the cache is enabled.

2.27 Verify Utility (VERIFY)

This section describes corrections to the Verify Utility.

2.27.1 File Header Identifier Error—Problem Corrected

V5.5-1 The Verify Utility mishandled file headers for files whose file identifier (FID) is a multiple of 64K. For example:

```
$ ANALYZE/DISK/REPAIR DKA300:
%ANALDISK-W-BADHEADER, file (65536,0,1) .;0
    invalid file header
-ANALDISK-I-BAD_STRUC_LEVEL, STRUCLEV field is bad
-ANALDISK-I-INVHEADER_BUSY, invalid file header marked "busy"
    in index file bitmap
```

Certain FIDs, which are multiples of 65536 (that is, 64K; FIDs whose low-order 16 bits are all 0), are considered reserved and are never used. These files are reserved to maintain compatibility with the RSX operating systems. VERIFY did not recognize these files as reserved.

System Manager Release Notes

2.27 Verify Utility (VERIFY)

When VERIFY passed over the INDEXF.SYS file, it did not treat these file headers as a special case and incorrectly marked them as invalid file headers. In doing so, VERIFY generated error messages that caused users to question the integrity of their data. If VERIFY was then invoked to repair the disk, it would also deallocate these reserved FIDs in the index file bitmap, allowing them to be allocated.

This problem has been corrected.

2.27.2 Lost File Header Repair Caused Corruption of Data—Problem Corrected

V5.5-1 A file system error was introduced in VMS Version 5.4 that sometimes caused the creation of lost extension file headers when a multiheader file was extended. After running ANALYZE/DISK_STRUCTURE/REPAIR on a disk with these lost file headers, VERIFY would leave many multiply allocated blocks on the disk.

The error was caused by VERIFY correctly finding lost extension headers and then putting them into the “lost” directory. But while doing this, VERIFY also attempted to update some dates in the ident area of these headers: the creation date, last modified date, and so on.

The ident area of a file header is a variable-length field and is shorter than normal in extension file headers. When VERIFY updated the date fields, it did so without checking the length of the ident area. The result was that VERIFY overwrote the map area of the file header. VERIFY now checks the length of the ident area, and the map area is no longer overwritten.

2.28 VAXstation 3100 Model 80 and VAXstation 4000 Model 60—Problem Corrected

V5.5-1 In previous versions of VMS, a problem existed on servers for VAXstation 3100 Model 80 and VAXstation 4000 Model 60 workstations that would cause a system failure every 11 minutes after memory errors until the system (or power) was turned off. The failure was a machine check caused by an unrecoverable memory error. In VMS Version 5.5-1, this problem has been corrected.

2.29 VMS Upgrade and VMS Update Procedures—Checking Directories

V5.5-2 When you apply a VMS upgrade or update procedure, you need to ensure that you move or rename any special test/debug files that you may have placed in any of the SYS\$SPECIFIC: or SYS\$SYSROOT: directories. This is because files that reside in these directories are used in place of files in SYS\$COMMON: directories.

At a minimum, you should check the following directories:

```
SYSSYSROOT:[SYSEXE]
SYSSYSROOT:[SYS$LDR]
```

Because the VMS upgrade and update procedures do not normally check or alter the contents of the SYS\$SYSROOT:[] directories, any files that you place into these directories for testing or debugging purposes will remain there unchanged until you remove or rename these files. If you do remove or rename these testing or debugging files, it may result in unpredictable system behavior.

2.30 VMS Volume Shadowing

For VMS Version 5.5–2, VMS Volume Shadowing (phase II) features controller performance assists to reduce copy and merge operation times on standalone systems and VAXcluster systems. The following sections describe the Version 5.5–2 shadowing performance enhancements.

Note

Refer to Section 2.30.6 before upgrading to VMS Version 5.5–2.

See also the *VMS Volume Shadowing Manual* for information about migrating phase I shadow sets to phase II. Digital intends to remove support for phase I in a future release of VMS. Additionally, OpenVMS Alpha will not support phase I shadowing.

2.30.1 Improved Performance for Merge and Copy Operations

There are two types of performance assist: the merge assist and the copy assist. The merge assist improves performance by using information maintained in controller-based write logs to merge only the data that is inconsistent across a shadow set. When a merge operation is assisted by the write logs, it is referred to as a “minimerge.” The copy assist reduces system resource usage and copy times by enabling a direct disk-to-disk transfer of data without going through VAX host node memory.

Assisted merge operations are usually too short to be noticeable. Phase II assisted copy operations are dramatically faster than either the phase I copy or phase II unassisted copy times. Improved performance is possible during the assisted copy operation because it consumes less CPU and interconnect resources. Although the primary purpose of the performance assists is to reduce the system resources required to perform a copy or merge operation, in some circumstances you may also observe improved read and write I/O performance.

Volume shadowing for VMS Version 5.5–2 supports both assisted and unassisted shadow sets in the same VAXcluster configuration. Whenever you create a shadow set, add members to an existing shadow set, or boot a system, the shadowing software reevaluates each device in the changed configuration to determine whether it is capable of supporting either the copy assist or the minimerge. Enhanced performance is possible only as long as all shadow set members are configured on controllers that support performance assist capabilities. If any shadow set member is connected to a controller without these capabilities, the shadowing software disables the performance assist for the shadow set.

The controllers supporting the assists are shown in Table 2–3.

Table 2–3 Controllers Supporting Copy and Merge Assists

	HSC50	HSC40	HSC60	HSC70	HSC90	KDM70	RF35	RF73
Merge Assist	N	Y	Y	Y	Y	Y	Y	Y
Copy Assist	N	Y	Y	Y	Y	N	N	N

System Manager Release Notes

2.30 VMS Volume Shadowing

The VMS Version 5.5-2 Volume Shadowing software will perform unassisted shadow set operations until you install the following:

- HSC software Version 6.5 or later
- KDM70 firmware Version 4.0 or later
- RF35 firmware Version T368 or later
- RF73 firmware Version T367 or later

The copy assist and minimerge are enabled by default, and are fully managed by the shadowing software.

Future versions of HSC50 software may support the merge assist. Support for the performance assists is not provided by any other storage controllers at this time.

2.30.2 Effects on Performance

The copy assist and minimerge are designed to reduce the time it takes to do copy and merge operations. In fact, you may notice significant time reductions on systems that have little or no user I/O occurring during the assisted copy or merge operation. Data availability is improved, too, because copy operations quickly make data consistent across the shadow set.

Minimerge Assist Performance Improvements

The phase II minimerge feature provides a very significant reduction in the elapsed time taken to perform merge operations. By using controller-based write logs, it is possible to avoid the total volume scan required by earlier merge algorithms and merge only those areas of the shadow set where write activity is known to be in progress.

Unassisted phase II merge operations often take several hours, depending on user I/O rates. Phase I merge operations typically take tens of minutes to complete, and negatively impact user I/O rates while in progress. Phase II minimerge operations typically complete in a few seconds, and are usually undetectable by users.

The exact time taken to complete a minimerge depends on the amount of outstanding write activity to the shadow set when the merge process is initiated, and on the number of shadow set members undergoing a minimerge simultaneously. Even under the heaviest write activity, a minimerge should complete in less than 1 minute. Additionally, minimerge operations consume minimal compute and I/O bandwidth.

Copy Assist Performance Improvements

Table 2-4 shows the approximate time required to perform copies on shadow sets that consist of two RA82s connected to an HSC70 controller on a VAX 8700 computer. The table shows assisted and unassisted copy times measured for one, two, three, and four concurrent copy operations where all source and target disks are on separate HSC requestors. For comparison purposes, the same test was performed using phase I shadow sets.

Note that the copy times in Table 2-4 reflect the best possible performance measurements taken on a controlled test system with no user I/O load. Copy times will vary according to each configuration, and generally will take longer on systems supporting user I/O. Performance benefits are also enhanced by having the source and target disks on different HSC requestors.

Table 2-4 Copy Times in Minutes for Two-Member RA82 Shadow Sets

Type of Shadowing	1 Set	2 Sets	3 Sets	4 Sets
Phase II: Assisted Copy	9	17	25	34
Phase II: Unassisted Copy with Identical Data	24	28	34	41
Phase II: Unassisted Copy with Different Data	91	120	157	200
Phase I Copy: ¹	15	20	21	22

¹With phase I shadowing, the HSC controller manages the copy operation.

Table 2-4 shows that for a single copy operation, a phase II assisted copy is significantly faster than any other. Note that as the number of simultaneous copies grows, the timing proportions change. This is caused by differing levels of parallelism in the various copying algorithms. Also, the overall copy times and their proportions will differ with the disk type, because each disk varies in total blocks and data transfer speed.

Additionally, Table 2-4 shows that unassisted phase II copies vary significantly in time, based on the similarity of data on the source and target disks. To make blocks with different data consistent, extra I/O operations are required. Thus, unassisted copy operations take significantly longer to make a shadow set consistent when the disk members contain different data. The phase II assisted copy operation does not depend on the similarity of data on the disks.

Finally, the measurements shown in Table 2-4 do not reflect the fact that multiple phase I copies negatively impact user I/O bandwidth. Phase II assisted copies allow significantly more user I/O than phase I even when multiple copies are performed simultaneously.

Refer to “Determining the DCD (Disk-Copy-Data) Connection Limit” in Section 2.30.4 for information about controlling the number of assisted copies an HSC controller can perform.

2.30.3 Enabling Assisted Merge Operations

Merge operations occur if a node fails or shuts down without dismounting its shadow sets. For unassisted merge operations, the shadowing software reads data from one member and compares it to the remaining members. If there are data inconsistencies, the shadowing software makes the data consistent across the shadow set. During a merge, shadowing methodically examines LBNs (logical block numbers) on each member until all LBNs have been compared and, if necessary, made identical.

The minimerge provides more efficient merge operation processing because it can take advantage of information about write operations that is logged in the controller memory. These write logs contain information about exactly which LBNs in the shadow set had write I/O requests outstanding (from a failed node). The remaining nodes use the write logs to merge those LBNs that are inconsistent across the shadow set.

System Manager Release Notes

2.30 VMS Volume Shadowing

Note

Because of the requirement to perform crash dump file consolidation, the shadowing software cannot perform a minimerge on a system disk. VMS Volume Shadowing manages the crash dump file consolidation and performs minimerge operations automatically.

How the Minimerge Assist Is Enabled

The minimerge operation can be enabled only on nodes running VMS Version 5.5-2. VMS Volume Shadowing automatically enables the minimerge if the controllers involved in accessing the physical members of the shadow set support it. See Table 2-3 for a list of supported controllers. Note that minimerge operations are possible even when shadow set members are connected to different controllers.

How the Minimerge Assist Is Disabled

VMS Volume Shadowing automatically disables minimerges if:

- A shadow set member is mounted on a controller that does not support the minimerge, on a controller running a version of firmware that does not support the minimerge, or on an HSC controller that has the assists disabled.
- The shadow set is mounted on a VAXcluster node that is not running VMS Version 5.5-2. In a VAXcluster system, each node with the shadow set mounted must be running VMS Version 5.5-2; otherwise, the minimerge is disabled for all nodes (including VMS Version 5.5-2 nodes) that have the shadow set mounted. See Section 2.30.6.2.
- The shadow set is mounted on a standalone system. For standalone systems, it is impossible to minimerge the shadow set because a single system reinitializes all disk controllers, thereby invalidating write logs during the boot process.

The following transient conditions can also cause a minimerge operation to be disabled:

- If a merge operation is already in progress when a node fails.
In this situation, the shadowing software cannot interrupt the merge operation with a minimerge.
- When there are not enough write log entries available in the controllers.
The number of write log entries available is determined by controller capacity. The shadowing software dynamically determines when there are enough entries to successfully maintain write I/O information. If the number of available write log entries is too low, shadowing temporarily disables logging in controller memory for that shadow set, polls the controllers while logging is disabled, and reenables logging when controller capacity is sufficient.
A controller retains a write log entry for each write I/O request until the write has been successfully completed to the virtual unit. This means that log entries associated with a virtual unit write cannot be reused or returned until all members of the shadow set have been updated.

The HSC controller, being a multiple unit controller, shares its write log entries between multiple disks. This pool of write log entries is statically allocated by the HSC controller and is managed by the shadowing software. If you want to ensure that write log entries are available so that shadowing is more likely to perform a minimerge instead of a merge operation, Digital

System Manager Release Notes 2.30 VMS Volume Shadowing

recommends you use the following guidelines for configuring HSC and KDM70 controllers:

- An HSC controller should have no more than 32 disks that are members of multiple-member shadow sets. You can observe the number of available write history entries (WHEs) on an HSC controller by entering the RUN VTDPY command at the HSC console. The VTDPY program displays a "Free List" of information, including the number of available WHEs. If the controller runs out of WHEs, shadowing will disable minimerges and perform unassisted merge operations.
- A KDM70 controller should have no more than 5 disks that are members of multiple-member shadow sets. As with HSC controllers, if the controller runs out of write log entries, shadowing will disable minimerges and perform unassisted merge operations.

These suggested numbers are general guidelines and will vary with the work load. Note that guidelines are not provided for RF-series devices; write log exhaustion does not typically occur with RF35 or RF73 disks because the write logs are not shared.

- When the controller write logs become inaccessible due to either of the following reasons, in which case the minimerge automatically reverts to an unassisted merge operation:
 - Controller failure causes write logs to be lost or deleted.
 - A device that is dual ported to multiple controllers fails over to its secondary controller. (If the secondary controller is capable of maintaining write logs, the minimerge operations will be reestablished quickly.)

2.30.4 Enabling Assisted Copy Operations

Copy operations occur when a disk is added to a shadow set to make the contents of the new member identical to that of the other members. Unassisted copy operations involve transferring data through the VAX CPU from the source disk to the target disk.

Unlike an unassisted copy, an assisted copy does not transfer data through the host node memory. The actual transfer of data is performed within the controller, thus the assisted copy decreases the impact on the system, the I/O bandwidth consumption, and the time required for copy operations. Shadow set members must be accessed from the same HSC controller in order to take advantage of the copy assist. The shadowing software controls the copy operation by using special MSCP copy commands to instruct the controller to copy specific ranges of LBNs. For an assisted copy, only one disk can be an active target for the copy at a time.

How the Copy Assist Is Enabled

By default, the VMS Volume Shadowing software and the HSC controller automatically enable the copy assist if the source and target disks are accessed through the same HSC controller. If you have disabled the assists on the HSC controller, see Section 2.30.5 for information about reenabling them.

How the Copy Assist Is Disabled

Shadowing automatically disables the copy assist if:

- The shadow set is mounted on an HSC controller that does not support the copy assist, or on an HSC controller with the copy assist disabled.
- A copy operation is initiated from a node running software earlier than VMS Version 5.5-2.

System Manager Release Notes

2.30 VMS Volume Shadowing

- The source and target disks are not accessed using the same HSC.
In the case of dual-ported HSC disks, it is possible to use the \$QIO SET PREFERRED PATH feature to force both disks to be accessed via the same HSC. This feature is useful because automatic failover in VAXcluster systems results in disks being accessed by either HSC controller. Note also when a dual-pathed HSC disk is dismounted, its access path will be automatically moved to the alternate HSC controller.
See the PREFER program in SYS\$EXAMPLES and refer to the *VMS I/O User's Reference Manual: Part I* for more information about setting a preferred path.
- The number of assisted copies specified by the HSC DCD (disk-copy-data) connection limit has been reached, at which point additional copies will be performed unassisted.

Determining the DCD (Disk-Copy-Data) Connection Limit

VMS Volume Shadowing implements the copy assist operation by issuing MSCP DCD commands to an HSC controller. You can control the number of assisted shadow set copies that can occur simultaneously on each HSC controller. This section describes how to determine a reasonable number based on how you prioritize your copy operations.

Note

Changes to the DCD connection limit are not necessary for most configurations. Only configurations that regularly perform more than four simultaneous copy operations (excluding failover) per HSC controller should consider altering the default setting.

For each assisted copy that is currently in progress, the HSC controller establishes an internal DCD connection between the source and target disks. By default, each HSC controller limits the number of DCD connections (and, therefore, the number of simultaneous assisted copies) to a maximum of four. The HSC DCD connection limit can be set from two to nine. If the setting is exceeded, then any request for an assisted copy will be refused by the HSC, and the copy will be performed unassisted.

You may need to change the DCD connection limit if:

- Copy operations are generally performed on disks with dissimilar data. In this case, unassisted copies take significantly longer than assisted copies. Therefore, setting a higher DCD connection limit would be advantageous.
- Absolute best copy times are desired. In many cases, a mix of assisted and unassisted copy operations can balance the work load over the entire system. The best mix of assisted and unassisted copies will vary based on the VAX computer, HSC controller, and disks involved, because each environment has a different performance profile. If you require an absolute minimum elapsed time for multiple simultaneous copies, then you may need to test different settings to find the best copy times for your configuration. Performance will also benefit if you ensure the source and target disks are not connected to the same HSC requestor. In this case, whether you raise or lower the DCD connection limit depends on your particular environment.

System Manager Release Notes 2.30 VMS Volume Shadowing

- User I/O performance has a higher priority than total elapsed time for the copy operations. Assisted copy operations conserve system I/O resources, which are then available for user I/O. Therefore, setting a higher DCD connection limit would be advantageous. However, note that large user I/O loads can extend the time required to perform an assisted copy operation because HSC controllers give priority to user I/O over DCD I/O.

Setting the Number of Copy Assists Per HSC

You can use the HSC command `SET SERVER DISK/DCD_CONNECTION_LIMIT` to control the mix of assisted and unassisted copies performed by a given HSC controller.

Follow these steps for each HSC controller on which you want to change the DCD connection limit:

1. Press `Ctrl/C` to get to the HSC prompt.
2. When the HSC prompt appears on the terminal screen, run the `SETSHO` program to change the limit. The following example sets the DCD connection limit to 6:

```
HSC> RUN SETSHO
SETSHO> SET SERVER DISK/DCD_CONNECTION_LIMIT=6
SETSHO-I Your settings require an IMMEDIATE reboot on exit.
SETSHO> EXIT
SETSHO-Q Rebooting HSC. Press RETURN to continue, CTRL/Y to abort:
```

After you issue these commands, the HSC controller automatically reboots.

```
INIPIO-I Booting...
```

With HSC Version 6.5 software, it is not possible to display the current DCD connection limit setting. This information will be displayed in a later HSC software release.

2.30.5 Controlling the Shadowing Performance Assists on HSC Controllers

To disable both the merge and copy performance assists on the HSC controller, follow these steps on each HSC controller for which you want to disable the assists:

1. Press `Ctrl/C` to get to the HSC prompt.
2. When the HSC prompt appears on the terminal screen, enter the following commands:

```
HSC> RUN SETSHO
SETSHO> SET SERVER DISK/NOHOST_BASED_SHADOWING
SETSHO-I Your settings require an IMMEDIATE reboot on exit.
SETSHO> EXIT
SETSHO-Q Rebooting HSC. Press RETURN to continue, CTRL/Y to abort:
```

After you issue these commands, the HSC controller automatically reboots.

```
INIPIO-I Booting...
```

To reenable the assists, follow the same procedure on your HSC controller except use the `/HOST_BASED_SHADOWING` qualifier on the `SET SERVER DISK` command.

With HSC Version 6.5 software, it is not possible to display whether the assists are enabled. This information will be displayed in a later HSC software release.

System Manager Release Notes

2.30 VMS Volume Shadowing

2.30.6 VMS Volume Shadowing Restrictions

This section describes known problems and other considerations that pertain to Volume Shadowing for VMS Version 5.5–2.

2.30.6.1 Recommended Version for VAXclusters, Volume Shadowing, and Striping

For VMS Version 5.5–2, nodes in a mixed-version VAXcluster system must run a minimum of VMS Version 5.4.

For mixed-version VAXcluster systems that use VMS Volume Shadowing, Digital recommends running a minimum of VMS Version 5.4–3.

VMS Version 5.5–2 requires Striping V2.0–007 when used in conjunction with VMS Volume Shadowing.

2.30.6.2 Using VMS Volume Shadowing in a Mixed-Version Cluster

The following restriction applies in mixed-version VAXcluster systems that use VMS Volume Shadowing with the VMS Version 5.5–2 shadowing performance assists.

A shadow set member that supports the minimerge assist may not be VMS MSCP served by multiple VAXcluster nodes that are running a mixture of Version 5.5–2 and previous versions of the VMS operating system.

This restriction must be observed in configurations where nodes access shadow set members through multiple VMS MSCP served paths.

Figure 2–1 shows two examples of VAXcluster systems (CI and DSSI) that are susceptible to violating this restriction.

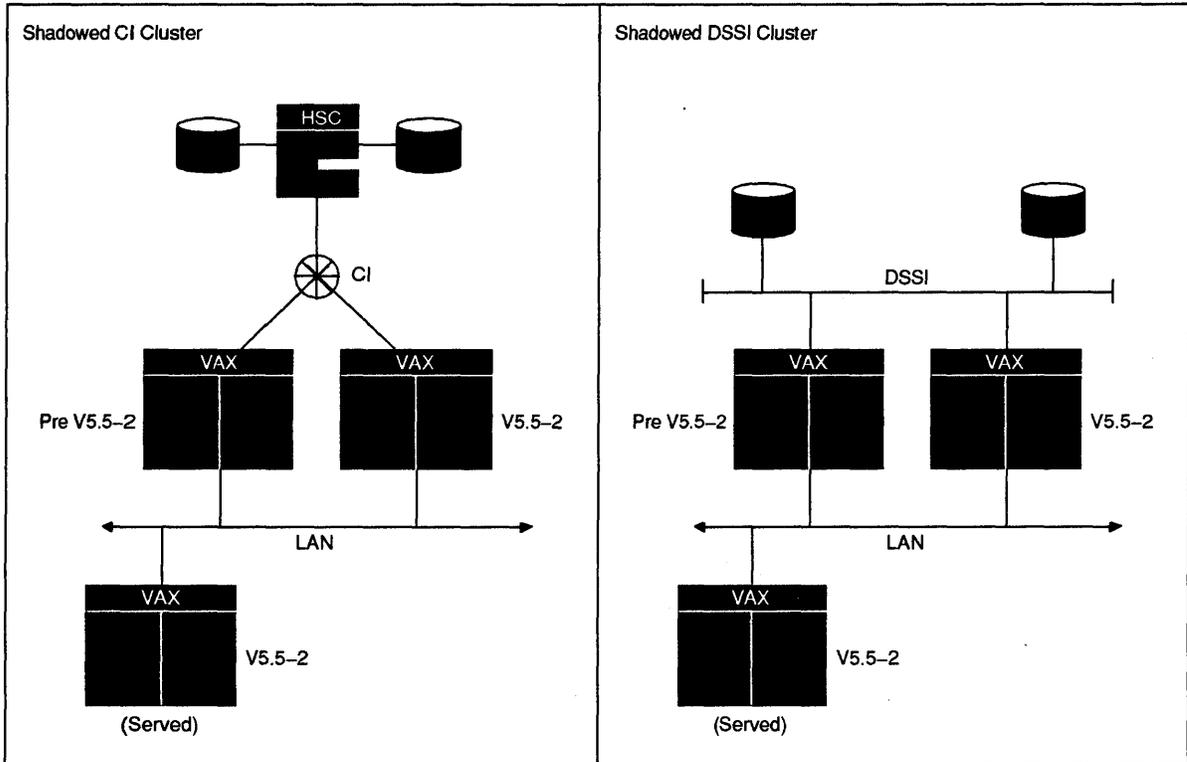
In Figure 2–1, when the served node accesses shadow set members through the node running Version 5.5–2, it will see the members as being able to support the minimerge feature. However, if the served node accesses the same shadow set members through the node running an earlier version of VMS, they will be flagged as not supporting the minimerge feature. This conflict regarding shadow set member status may cause a system failure.

The following configuration modifications can be used to overcome this restriction:

- Ensure that all nodes serving HSC and RF-series disks that are shadow set members run VMS Version 5.5–2.
- Ensure that VMS Version 5.5–2 nodes do not share a common allocation class with nodes running previous versions. This will require changes to HSC and RF disk controller allocation classes. See the *VMS Volume Shadowing Manual* and the *VMS VAXcluster Manual* for information about setting allocation classes.
- Disable the phase II shadowing performance assists at the HSC console following the instructions in Section 2.30.5. This disables both the merge and copy assists.
- Disable VMS MSCP serving using the system parameter `MSCP_LOAD` on either the V5.5–2 nodes or the nodes running earlier versions.

To reenble the performance assists, see Section 2.30.5.

Figure 2-1 Serving Nodes in a CI or DSSI Configuration



ZK-5234A-GE

2.30.6.3 Memory Requirements for VMS Volume Shadowing

VMS Volume Shadowing (phase II) uses one additional process, SHADOW_SERVER, on nodes that have shadowing enabled. This section includes a breakdown of the additional memory requirements for VMS Volume Shadowing Version 5.5-2. Note that the following list is not exhaustive; it contains only major memory consumers.

- Shadowing (phase II) code:
 - SHDRIVER—137 pages (about 70K bytes)
 - SHADOW_SERVER—the maximum working set size of the server process is:

$$PQL_DWSQUOTA + (128 \times SHADOW_MAX_COPY)$$
 Using default values for these parameters, this value is 712. A typical working set size for this process is 150 to 200 pages.
- Shadowing (phase II) data:
 - One SHAD structure (2 pages) per shadow set
 - One additional VCB (volume control block) (240 bytes, about 1/2 page) per shadow set for the virtual unit.
 - One additional UCB (unit control block) (164 bytes, about 1/3 page) per shadow set for the virtual unit.

System Manager Release Notes

2.30 VMS Volume Shadowing

Note that the UCBs for shadow set members are already present if the disks are visible on the system, and the VCBs for the members are created whenever the disks are mounted, regardless of shadowing status.

- I/O operations performed:
 - One additional IRP per *outstanding* read (an IRP is 176 bytes, or about 1/3 page).
 - For an n -member shadow set, n additional IRPs per *outstanding* write. (One IRP is needed for each of the shadow set members, and one IRP is needed for the virtual unit. For an unshadowed disk, only one IRP is needed for an outstanding write; for an n -member shadow set, $n + 1$ IRPs are needed, thus the difference of n .)
 - Additional IRPs are used for various other operations; for example, if the disk is in merge state, one additional IRP per member is needed.
 - Buffer space during copies and merges (approximately 100 pages per copy or merge stream).
 - Each physical device using minimerge write logs requires a maximum additional 8K bytes of nonpaged pool.

2.30.6.4 Unnecessary Merge During System Reboot—Suggested Workaround

In a VAXcluster system, when a node that has a shadow set mounted shuts down incorrectly (or crashes), a remaining node will merge the shadow set. If the merge operation completes and the shadow set is either subsequently dismounted or the entire cluster is shut down, a remount of the shadow set may result in an unnecessary merge operation.

To avoid the unnecessary merge operation, ensure that a file system rebuild operation (for example, MOUNT/REBUILD) is performed on the shadow set prior to dismount or shutdown.

2.30.6.5 Connection Loss During Shadowing State Change May Cause Bugcheck

In the rare event that multiple connection losses occur during a shadow set state change (such as when a disk member is being dismounted), a system bugcheck may result.

The bugcheck occurs when the connection loss prevents shadowing software from updating the storage control block information on member disks after the local update has been committed. This is reported by the following message:

```
SHADDETINCON, SHADOWING detects inconsistent state
```

This problem will be fixed in a future VMS release.

2.30.7 Supported Shadow Sets Maximum Increased to 75

With VMS Version 5.5, VMS Volume Shadowing (phase II) supports a maximum of 75 shadow sets on a single node or in a VAXcluster system. Prior to VMS Version 5.5, phase II shadowing supported up to 32 shadow sets. The number of shadow sets supported is independent of controller and device types.

2.30.8 SHOW DEVICES Display Incorrect for Merge Copy

This note pertains to VMS Volume Shadowing (phase II) only. On rare occasions, the SHOW DEVICES command might indicate the status of a merge operation as being 100 percent merged when, in fact, the merge is still in progress.

This is a problem with the SHOW DEVICES display only. The VMS Volume Shadowing (phase II) merge operation is functioning correctly. This problem will be fixed in a future version of VMS.

2.30.9 Assisted Copy Operation Resets Incorrectly After Minimerge

VMS Volume Shadowing (phase II) may incorrectly reset an assisted copy operation after a system crash and a subsequent minimerge operation. If a node with the shadow set mounted crashes at the time an assisted copy operation is in progress, the copy may restart at 0 percent copied.

The expected behavior is for the copy operation to continue at the percentage copied when the crash occurred. For example, if the shadow set was 33 percent copied at the time of the crash, the copy should resume at 33 percent after the system crash and the minimerge completes.

This problem will be fixed in a future version of VMS.

2.31 VMS Version 5.4 to Version 5.5 Queue Database Upgrade Problem Resolution

V5.5-2 This section describes a workaround for a problem when you upgrade the queuing system from VMS Version 5.4 or earlier to the new Version 5.5 design. If your system is currently running VMS Version 5.5, this information does not apply to you. This problem is not an issue for versions of VMS following VMS Version 5.5.

2.31.1 Problem Information

During phase 6 of the VMS Version 5.5 upgrade, you cannot convert to the new queue manager because the queue conversion fails no matter how the following question is answered:

```
"Do you wish to upgrade to the new JOB_CONTROL at this time [NO]?"
```

If you answer NO, you will not upgrade to the new job controller. If you answer YES, the job controller starts, but conversion of the queues fails with the following messages:

```
Starting the new job controller...
```

```
Starting the new queue manager...
```

```
%JBC-E-QMANDEL, unexpected queue manager process termination  
-SYSTEM-F-PROTINSTALL, protected images must be installed  
%JBC-E-QMANDEL, unexpected queue manager process termination  
-SYSTEM-F-PROTINSTALL, protected images must be installed  
%JBC-E-QMANNOTSTARTED, queue manager could not be started
```

The queue conversion fails because the IPC\$SHARE image is not installed.

System Manager Release Notes

2.31 VMS Version 5.4 to Version 5.5 Queue Database Upgrade Problem Resolution

Problem Resolution

To work around this problem, perform the following steps:

1. Before you upgrade to VMS Version 5.5, issue the following command to display all queues and jobs in the system:

```
$ SHOW QUEUE/ALL/FULL/OUTPUT=QUEUES.TXT
```

If some of the queues have been corrupted, you can notify users if their jobs are not transferred during the upgrade. Likewise, the following commands may also be useful:

```
$ SHOW QUEUE/FORM/FULL/OUTPUT=QFORMS.TXT
```

```
$ SHOW QUEUE/CHARACTERISTIC/FULL/OUTPUT=QCHAR.TXT
```

Corruption issues are resolved in the new queuing system. Therefore, queuing system database corruption will not be an issue in future releases of VMS.

2. During phase 6 of the VMS Version 5.5 upgrade, answer NO to the following question:

```
"Do you wish to upgrade to the new JOB_CONTROL at this time [NO]?"
```

3. Let the upgrade run to completion. AUTOGEN will run and the system will reboot. Once the system is rebooted, the IPC\$SHARE image is installed.
4. Log in to the SYSTEM account.
5. Make sure the disks are mounted. Disks holding queue database files and disks holding users' files must be available to start the new queue manager and to restore batch and print jobs that were previously submitted.
6. Disable the job controller's "cold start" verification of its database as follows:

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> SET JOBCTLD 1
SYSGEN> WRITE ACTIVE
SYSGEN> EXIT
```

Setting this undocumented system parameter allows the data in the former queue database (JBCSYSQUE.DAT) to be saved for transfer to the VMS Version 5.5 queue database (even if corruption exists in the former database). You can use the System Generation Utility (SYSGEN) to change this parameter, because it is an isolated and temporary change.

7. Redefine any queues, forms, and characteristics that may not be transferred due to corruption. See step 1.

If there is minor corruption in the JBCSYSQUE.DAT file, the uncorrupted data will be transferred to the new database. If there is major corruption in the JBCSYSQUE.DAT file, the transfer of data to the new database might fail.

8. Execute the following command procedure:
SYS\$UPDATE:VMS\$UPGRADE_A55_V55.COM.
9. Answer YES to the following question:

```
"Do you wish to convert queues now ? (YES,NO,ABORT)" YES
```

2.31 VMS Version 5.4 to Version 5.5 Queue Database Upgrade Problem Resolution

Answering YES upgrades your system to VMS Version 5.5¹ with the new job controller and queue manager. It also transfers information from the former queue database (JBCSYSQUE.DAT) to the new queue database. When you enter the START/QUEUE/MANAGER command, the new queue manager is started.

Answering NO upgrades your system to VMS Version 5.5 with the new job controller and queue manager, but it does not transfer information from the former queue database (JBCSYSQUE.DAT) to the new queue database. When you start the new queue manager with the START/QUEUE/MANAGER command, the new queue database will be empty (that is, it will not contain any information about queues or jobs).

Answering ABORT leaves your system as VMS Version A5.5. Your system continues to run the former job controller and queue manager and uses the former queue database (JBCSYSQUE.DAT).

Caution

The new queue manager is in a stopped state upon completion of the upgrade. If you do answer YES to the queue conversion question, do not include the /NEW qualifier on the START/QUEUE/MANAGER command; this qualifier effectively erases the queuing database. For information about the queuing system, see the *Guide to Maintaining a VMS System*.

2.32 XQP and File System in Mixed-Version VAXclusters

V5.5-2

A software correction in the VMS Version 5.5-2 XQP and File System may increase the chances of applications getting an unexpected error status (SS\$_DEADLOCK) if the VMS Version 5.5-2 system is in a VAXcluster with systems running earlier versions of VMS. Digital recommends that you complete the update to VMS Version 5.5-2 on all VAXcluster nodes as quickly as possible to minimize the possibility of user applications receiving this error. See Section 3.15.5 for more details.

¹ VMS Version V5.5, V5.5-1, and V5.5-2 use the new queue manager (process name QUEUE_MANAGER) and job controller (process name JOB_CONTROL); VMS Version A5.5, A5.5-1, and A5.5-2 use the previous queue manager, which is a function of the previous job controller (process name JOB_CONTROL).

Programmer Release Notes

This chapter contains information about VMS Version 5.5-2 for application and system programmers.

3.1 DSA Multiple Controllers Timeout Errors—Problem Corrected

V5.5-1 Configurations that have multiple DSA (DIGITAL Storage Architecture) controllers that are idle for extended periods of time can log false host access timeout errors. When multiple DSA controllers are configured on one bus, more time is needed to poll inactive controllers to prevent false access timeout errors from occurring. More time has been added to allow for these configurations.

3.2 LATSVM Error During Hangup Event—Problem Corrected

V5.5-1 LATSVM could have failed with an access violation error when a LAT device it was controlling received a hangup event. This may have occurred if the symbiont process was driving multiple queues. In VMS Version 5.5-1, LATSVM no longer causes an access violation error. It now correctly sets the queue status bits during a hangup event.

3.3 Mail Utility (MAIL)—Callable Mail No Longer Loses System Privileges

V5.5-2 In previous versions of VMS, if you ran a program that used callable mail and you set the SYSPRV privilege, the program failed, even though the program worked correctly without SYSPRV.

In VMS Version 5.5-2, this problem has been corrected; having SYSPRV privilege set in the current privilege mask upon entry into a program that uses callable mail no longer causes the program to fail.

3.4 Mailbox Driver EXE\$SNDEVMSG Routine Error

V5.5-1

Note

This problem is mentioned in the *Cover Letter Supplement for VMS Version 5.5* (part number AV-PMX6A-TE) under the heading "Mail Driver Problem and ACMS."

If you had ACMS installed on your system running VMS Version 5.5, you were unable to log in to ACMS through controlled terminals except for LAT-dedicated service ports. On other controlled terminals, after you started the ACMS terminal subsystem, you would have seen the "Connected to ACMS" and "Press <RET> to continue . . ." messages, but you would not get any response from ACMS due to the problem with the mailbox driver.

Programmer Release Notes

3.4 Mailbox Driver EXE\$SNDEVMSG Routine Error

An error in the mailbox driver's EXE\$SNDEVMSG routine caused the driver to write a string count one byte longer than the actual string. If the application expected a \$QIO read request to return data sent by EXE\$SNDEVMSG, and the application does one or more of the following, the extra byte caused a problem:

- Created the mailbox with MAXMSG set to the exact size of the string that it expected EXE\$SNDEVMSG to write
- Used the length returned by the \$QIO read request to, for example, compare strings or simply to verify that the length was an expected value

VMS Version 5.5-1 enables you to log into ACMS through all controlled terminals.

3.5 Pseudoterminal Driver Control Connection Routines

V5.5-2 In previous versions of VMS, the following restrictions were in effect for the pseudoterminal driver control connection routines PTD\$READ, PTD\$READW, and PTD\$WRITE:

- The maximum buffer size permitted was 508 bytes.
- Read or write buffers could not span a page.

In VMS Version 5.5-2, these restrictions have been removed. For instance, the maximum buffer size is now is as much memory as you decide to allocate.

3.6 Remote Port Command Procedure Sample

V5.5-2 The following is a previously undocumented sample command procedure that provides remote ports with a means of accessing and controlling a modem.

```
$!      RMT_PORT.COM
$!      Example Remote Port Procedure - V1.0
$!
$ port  := tta2:
$ baud  := 2400
$!
$ alloc 'port'
$ set term 'port' /modem /speed='baud' /noautobaud
$ deassign sys$input
$ define sys$input sys$command
$ set host /dte 'port'
$ deassign sys$input
$ dealloc 'port'
```

3.7 RMS Journaling

The release notes in this section pertain to the RMS Journaling Services.

3.7.1 Access to a Remote Indexed Read-Only File Can Fail—Problem Corrected

V5.5-2 A remote read-only indexed file with RMS Recovery Unit Journaling enabled and with a XABKEY data structure linked into the XAB list received the following error when the file was opened:

```
RMS-E-DDTM_ERR, error from RMS
```

In VMS Version 5.5-2, this problem has been corrected.

3.7.2 Deadlock Among Multiple Detached Recovery Servers—Problem Corrected

V5.5-2 In previous versions of VMS, a deadlock could occur when an attempt to open a journal failed while multiple detached recovery servers tried to coordinate access to recover a file. The attempt to open the journal could fail if the device was not mounted or if some other resource problem occurred.

In VMS Version 5.5–2, this problem has been corrected.

3.7.3 RMS Journaling Detached Recovery—Problem Corrected

V5.5-2 In previous versions of VMS, when you used the RMS Journaling Services, Dynamic Memory Exhausted (RMS\$_DME) errors caused the recovery server to fail. In VMS Version 5.5–2, the maximum number of threads active in a single RMS Journaling recovery process has been lowered to 30, which prevents these errors from occurring.

3.7.4 TID Format in Transaction Error Messages

V5.5-2 In VMS Version 5.5–2, on failure of transaction commit or abort, RMS dumps the Transaction Identification (TID) into an error message, rather than formatting it in a standard way. If the bytes of the TID are labeled from highest byte address (P) to lowest byte address (A), as follows:

```
PONMLKJIHGFEDCBA
```

Then the RMS TID display is:

```
DCBA HGFE LKJI PONM
```

So the conversion from one format to the other remains straightforward, the current standard format is:

```
DCBA-FE-HG-IJ-KLMNOP
```

3.8 Screen Management Run-Time Library (SMGRTL)

The release notes in this section pertain to the Screen Management Run-Time Library (SMGRTL).

3.8.1 BLOCK_BORDER—Problem Corrected

V5.5-2 In previous versions of VMS, when you changed a virtual display's border to BLOCK_BORDER, the border disappeared. In VMS Version 5.5–2, this problem has been corrected.

3.8.2 Incorrect Output with Terminal Device Set to VT80—Problem Corrected

V5.5-2 In previous versions of VMS, SMG\$ output was incorrect when a terminal device was set to VT80. In VMS Version 5.5–2, this problem has been corrected.

3.8.3 Input Routine—Problem Corrected

V5.5-2 In previous versions of VMS, the SMG\$ virtual keyboard remained locked when you ran a \$UNWIND through an SMG\$ input routine. In VMS Version 5.5–2, this problem has been corrected.

Programmer Release Notes

3.8 Screen Management Run-Time Library (SMGRTL)

3.8.4 Line Wrapping Outside Virtual Display—Problem Corrected

V5.5-2 In previous versions of VMS, line wrapping in an area outside a defined scrolled region of an SMG virtual display placed the wrapped text inside the scrolled region, even if the scrolled region was far from where the original text was placed. In VMS Version 5.5-2, this problem has been corrected.

3.8.5 SMG\$ Viewport—Problem Corrected

V5.5-2 In previous versions of VMS, if you created an SMG\$ viewport on a virtual display with wide characters and defined the viewport so that the leftmost wide characters could not be seen, twice the number of wide characters were dropped from the display. In VMS Version 5.5-2, this problem has been corrected.

3.8.6 SMG\$ Access Violation—Problem Corrected

V5.5-2 In previous versions of VMS, when SMG\$ parsed a VT52 cursor addressing sequence for the home position, an access violation occurred. In VMS Version 5.5-2, the problem has been corrected.

3.8.7 SMG\$ Virtual Display Access Violation—Problem Corrected

V5.5-2 In previous versions of VMS, an SMG\$ virtual display having a vertical label on either side of the virtual display and pasted on the pasteboard with a row number less than 1 caused an access violation. In VMS Version 5.5-2, this problem has been corrected. The portion of the virtual display that is visible on the pasteboard is shown properly.

3.8.8 SMG\$CHANGE_PBD_CHARACTERISTICS Routine with DECwindows DECterm—Problem Corrected

V5.5-2 In previous versions of VMS, the SMG\$CHANGE_PBD_CHARACTERISTICS routine did not work correctly when you used it on a DECwindows DECterm created by the SMG\$CREATE_PASTEBOARD routine. In VMS Version 5.5-2, this problem has been corrected.

3.8.9 SMG\$CHANGE_RENDITION Display—Problem Corrected

V5.5-2 In previous versions of VMS, when you pasted a virtual display onto the pasteboard with a row number less than 1, calling the SMG\$CHANGE_RENDITION routine did not change the rendition when the specified region was partially on the screen. In VMS Version 5.5-2, this problem has been corrected.

3.8.10 SMG\$CHANGE_VIEWPORT Label Disappearance—Problem Corrected

V5.5-2 In previous versions of VMS, when you changed a viewport with a special graphics label, calling the SMG\$CHANGE_VIEWPORT routine caused the label to disappear. In VMS Version 5.5-2, this problem has been corrected.

3.8.11 SMG\$CHANGE_VIRTUAL_DISPLAY—Problems Corrected

V5.5-2 In previous versions of VMS, calling the SMG\$CHANGE_VIRTUAL_DISPLAY routine caused an access violation if a viewport was defined completely beyond the boundary of the virtual display. In VMS Version 5.5-2, this problem has been corrected.

Also, when you used the SMG\$CHANGE_VIRTUAL_DISPLAY routine on a virtual display with a label containing special graphics characters, the label disappeared. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes
3.8 Screen Management Run-Time Library (SMGRTL)

3.8.12 SMG\$CREATE_MENU Access Violation—Problem Corrected

V5.5-2 In previous versions of VMS, calling the SMG\$CREATE_MENU routine with no choices specified resulted in an access violation. In VMS Version 5.5-2, this problem has been corrected.

3.8.13 SMG\$DISABLE_BROADCAST_TRAPPING Incompatible with SMG\$DISABLE_UNSOLICITED_INPUT—Problem Corrected

V5.5-2 In previous versions of VMS starting with VMS Version 5.4, if a user disabled both broadcast trapping and unsolicited input with the SMG\$DISABLE_BROADCAST_TRAPPING and the SMG\$DISABLE_UNSOLICITED_INPUT routines, you could not turn these functions back on. In VMS Version 5.5-2, this problem has been corrected.

3.8.14 SMG\$DISABLE_UNSOLICITED_INPUT Invalid Channel Error—Problem Corrected

V5.5-2 In previous versions of VMS, calling the SMG\$DISABLE_UNSOLICITED_INPUT routine from an AST routine triggered by the SMG\$ENABLE_UNSOLICITED_INPUT routine generated an “invalid channel” error. In VMS Version 5.5-2, this problem has been corrected.

3.8.15 SMG\$EXECUTE_COMMAND Aborted by STOP/ID=xxxx—Problem Corrected

V5.5-2 In previous versions of VMS, the SMG\$EXECUTE_COMMAND routine hung if the process was aborted by STOP/ID=xxxx. In VMS Version 5.5-2, this problem has been corrected.

3.8.16 SMG\$INSERT_CHARS Wrap—Problem Corrected

V5.5-2 In previous versions of VMS, the SMG\$INSERT_CHARS routine wrapped incorrectly if you also specified the SMG\$M_WRAP_WORD routine. In VMS Version 5.5-2, this problem has been corrected.

3.8.17 SMG\$INVALIDATE_DISPLAY Wide Character Display—Problem Corrected

V5.5-2 In previous versions of VMS, when you invalidated a virtual display containing a line of wide characters by calling the SMG\$INVALIDATE_DISPLAY routine, the line of wide characters was drawn incorrectly when the virtual display was redrawn. In VMS Version 5.5-2, this problem has been corrected.

3.8.18 SMG\$M_RETURN_IMMED Flag—Problem Corrected

V5.5-2 In previous versions of VMS, when you set the flag in the SMG\$M_RETURN_IMMED routine in a call to the SMG\$SELECT_FROM_MENU routine, the PF1 key did not return immediately. Instead it waited for another character to be entered (as if it were still functioning as a GOLD key) and then always returned 0 as the terminator code. In VMS Version 5.5-2, this problem has been corrected.

3.8.19 SMG\$PASTE_VIRTUAL_DISPLAY Screen Output—Problem Corrected

V5.5-2 In previous versions of VMS, the SMG\$PASTE_VIRTUAL_DISPLAY routine did not always correctly display occlusions, resulting in incorrect screen output. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes

3.8 Screen Management Run-Time Library (SMGRTL)

3.8.20 SMG\$PRINT_PASTEBOARD Routine—Problem Corrected

V5.5-2 In previous versions of VMS, it was possible to receive a return status code of 0000803A when you called the SMG\$PRINT_PASTEBOARD routine. This is not a valid return status code, it is only the low-order word of the correct return status code. In VMS Version 5.5-2, this problem has been corrected; the entire return status code is returned to the caller.

3.8.21 SMG\$PUT_CHARS_MULTI Routine Rendition Argument—Problem Corrected

V5.5-2 In previous versions of VMS, an access violation occurred when the optional **rendition-set** argument was omitted when you called the SMG\$PUT_CHARS_MULTI routine. In VMS Version 5.5-2, this problem has been corrected.

3.8.22 SMG\$PUT_CHARS_WIDE and SMG\$PUT_CHARS_HIGHWIDE Text Overwritten—Problem Corrected

V5.5-2 In previous versions of VMS, the cursor position was not properly updated after you called the SMG\$PUT_CHARS_WIDE routine or the SMG\$PUT_CHARS_HIGHWIDE routine. This caused a second call to the routine to overwrite the text written by the previous call. In VMS Version 5.5-2, this problem has been corrected.

3.8.23 SMG\$PUT_CHARS_WIDE ASCII Characters Displayed—Problem Corrected

V5.5-2 In previous versions of VMS, when you called the SMG\$PUT_CHARS_WIDE routine with special graphics specified as the default character, the set still had the output displayed as ASCII characters. In VMS Version 5.5-2, this problem has been corrected.

3.8.24 SMG\$PUT_CHARS_WIDE and SMG\$PUT_CHARS_HIGHWIDE Start-Row and Start-Column Arguments—Problem Corrected

V5.5-2 In previous versions of VMS, if you called the SMG\$PUT_CHARS_WIDE or the SMG\$PUT_CHARS_HIGHWIDE routine and specified either the **start-row** or the **start-column** argument but not both, then neither specified value was used and the cursor did not change from its previous position. In VMS Version 5.5-2, this problem has been corrected.

3.8.25 SMG\$PUT_HELP_TEXT with INVISIBLE Attribute—Problem Corrected

V5.5-2 In previous versions of VMS, when you called the SMG\$PUT_HELP_TEXT routine with the INVISIBLE attribute, prompt strings such as "Topic?" or "XXX Subtopic?" were displayed, but other outputs were not displayed. Also, if the next help screen had the same characters as these prompt strings in the same location, the characters remained unerased. These characters should have been erased, because the display was designated as being invisible. In VMS Version 5.5-2, this problem has been corrected.

3.8.26 SMG\$PUT_LINE Output—Problem Corrected

V5.5-2 In previous versions of VMS, the number of lines specified by the LINE_ADV parameter was output after the output line during a call to the SMG\$PUT_LINE routine. However, when the output reached the bottom of a scrolled region, the behavior changed to outputting LINE_ADV number of lines before the output line. This changed the way formatted output appeared. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes
3.8 Screen Management Run-Time Library (SMGRTL)

3.8.27 SMG\$PUT_LINE_HIGHWIDE Routine—Problem Corrected

V5.5-2 In previous versions of VMS, SMG\$PUT_LINE_HIGHWIDE did not properly clear the remainder of the line when overwriting previously written highwide characters. In VMS Version 5.5-2, this problem has been corrected.

3.8.28 SMG\$PUT_LINE_MULTI Routine—Problems Corrected

V5.5-2 In previous versions of VMS, when you called the SMG\$PUT_LINE_MULTI routine a second time, after a call in which the output text exceeded the right edge of the display, only one character was output. In VMS Version 5.5-2, this problem has been corrected.

Also, if you called the SMG\$PUT_LINE_MULTI routine immediately after you called the SMG\$PUT_LINE routine with a string exactly the width of the virtual display, the SMG\$PUT_LINE_MULTI routine would not output any text. In VMS Version 5.5-2, this problem has been corrected.

Finally, word wrapping did not work properly when you called the SMG\$PUT_LINE_MULTI routine. In VMS Version 5.5-2, this problem has been corrected.

3.8.29 SMG\$PUT_STATUS_LINE Display Character—Problem Corrected

V5.5-2 In previous versions of VMS, when you called the SMG\$PUT_STATUS_LINE routine with *n* characters worth of text on an *n* column terminal, only *n-1* characters appeared.

For example, an output of 80 characters on an 80 column display with SMG\$PUT_STATUS_LINE resulted in an output of only 79 characters. In VMS Version 5.5-2, this problem has been corrected.

3.8.30 SMG\$READ_COMPOSED_LINE Routine—Problems Corrected

V5.5-2 The following problems have been corrected in the SMG\$READ_COMPOSED_LINE routine:

- In previous versions of VMS, when you called the SMG\$READ_COMPOSED_LINE routine, only the portion of the screen occupied by the input string was updated in the SMG display storage. If the initial string was longer than the resultant string, then the portion of the display storage that was updated was too short to show that the deleted characters were changed to be blank. In VMS Version 5.5-2, this problem has been corrected.
- In previous versions of VMS, SMG\$READ_COMPOSED_LINE terminated when you pressed a key defined with an equivalence string. In VMS Version 5.5-2, this problem has been corrected.
- In previous versions of VMS, when you pressed a key that has been predefined, the SMG\$READ_COMPOSED_LINE routine did not accept the number of characters that could be fit in the display field and return the buffer full terminator code (SMG\$K_TRM_BUFFER_FULL). The SMG\$READ_COMPOSED_LINE routine returned the status code SS\$_BADPARAM and did not return any of the text that you entered before you pressed the predefined key. In VMS Version 5.5-2, this problem has been corrected.
- In previous versions of VMS, when you pressed Ctrl/Z as a response to the SMG\$READ_COMPOSED_LINE routine, it was echoed as *EXIT*, but the *EXIT* was not properly entered into the SMG internal screen memory. As a result, when you updated the virtual display, or otherwise caused the *EXIT*

Programmer Release Notes

3.8 Screen Management Run-Time Library (SMGRTL)

to be removed, the minimum update procedure did not remove the *EXIT*. In VMS Version 5.5-2, this problem has been corrected.

- In previous versions of VMS, if you used the SMG\$READ_COMPOSED_LINE routine on the last line of a virtual display occupying the bottom of the screen, then the entire screen incorrectly scrolled up one line when the virtual display being read was partially occluded by another virtual display. In VMS Version 5.5-2, this problem has been corrected.
- In previous versions of VMS, calling the SMG\$READ_COMPOSED_LINE routine with an equivalence string defined for a specific key caused truncation of the output due to lack of space in the input buffer for the equivalence string. This resulted in the permanent truncation of the equivalence string displayed and the read operation immediately terminated. In VMS Version 5.5-2, this problem has been corrected.

3.8.31 SMG\$READ_STRING Routine—Problems Corrected

V5.5-2

In previous versions of VMS, an LF character echoed when you called the SMG\$READ_STRING routine with the TRM\$M_TM_TRMNOECHO modifier specified. In VMS Version 5.5-2, this problem has been corrected.

Also, the SMG\$READ_STRING routine no longer showed a prompt on the next line when you pressed a keypad key. In VMS Version 5.5-2, this problem has been corrected.

3.8.32 SMG\$READ_VERIFY Input_Length—New Optional Argument

V5.5-2

In VMS Version 5.4, the run-time library SMG\$READ_VERIFY routine returned only those characters entered in the **resultant-string** argument. This created a problem when you specified an initial string and then pressed Return. Pressing Return caused the SMG\$READ_VERIFY routine to return a null string instead of the initial string, since you did not actually enter any characters. In VMS Version 5.5-2, SMG\$READ_VERIFY returns all of the characters in the input buffer, including the initial string.

To accommodate users who took advantage of the previous behavior, VMS Version 5.5-2 includes a new optional argument (**input_length**) for the SMG\$READ_VERIFY routine. The **input_length** argument is an unsigned word passed by reference. This argument returns the actual number of characters entered.

With VMS Version 5.5-2, the complete calling format for SMG\$READ_VERIFY is as follows:

Programmer Release Notes

3.8 Screen Management Run-Time Library (SMGRTL)

```
RET_STATUS.wlc.v = SMG$READ_VERIFY
KEYBOARD_ID.rl.r,
OUT_STRING.wt.dx,
IN_STRING.rt.dx,
PIC_STRING.rt.dx
[, [FILL_CHAR.rl.r]
[, [TIMEOUT.rl.r]t.dx]
[, [PROMPT_STRING.rt.dx]
[, [MODIFIERS.r [, [CLEAR_CHAR.rt.dx]
[, [TIMEOUT.rl.r]
[, [TERMINATOR_SET.rt.ds]
[, [INI_OFFSET.rl.r]
[, [TERMINATOR_CODE.wwu.r]
[, [DISPLAY_ID.rl.r]
[, ALT_ECHO_STRING.rt.dx,
[, ALT_DISPLAY_ID.rl.r]
[, ALT_ECHO_STRING.rt.dx,
[, ALT_DISPLAY_ID.rl.r]
[, [RENDITION_SET.rl.r]
[, [RENDITION_COMPLEMENT.rl.r]
[, [INPUT_LENGTH.wwu.r]
```

3.8.33 SMG\$READ_VERIFY Routine Resultant-String Argument—Problem Corrected

V5.5-2 Starting with VMS Version 5.4, the **resultant-string** argument in SMG\$READ_VERIFY returned only those characters actually entered by the user and ignored any extra characters that were supplied by the **initial_string** argument. In all previous versions, the entire string of input characters entered by the user as well as those supplied by the initial string were returned. With VMS Version 5.5-2, SMG returns to the behavior prior to VMS Version 5.4.

To have SMG\$READ_VERIFY return the actual number of characters entered, use the new optional argument **input_length**.

3.8.34 SMG\$REMOVE_LINE Line Characters—Problem Corrected

V5.5-2 In previous versions of VMS, when you called the SMG\$REMOVE_LINE routine to remove a line containing both line drawing characters and normal characters, the rendition of the characters remained unchanged. This caused any other characters written to the same location as the previous line drawing characters to display as line drawing characters. In VMS Version 5.5-2, this problem has been corrected.

3.8.35 SMG\$SAVE_VIRTUAL_DISPLAY Special Graphics Display—Problem Corrected

V5.5-2 In previous versions of VMS, after you saved the display of special graphics to a file using SMG\$SAVE_VIRTUAL_DISPLAY, SMG\$LOAD_VIRTUAL_DISPLAY should have been able to read the file and create the same virtual display. Instead, it returned SMG\$_INVARG. In VMS Version 5.5-2, this problem has been corrected.

3.8.36 SMG\$SNAPSHOT Routine—Problem Corrected

V5.5-2 In previous versions of VMS, the routine SMG\$SNAPSHOT could hang when used on a hardcopy terminal. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes

3.8 Screen Management Run-Time Library (SMGRTL)

3.8.37 SMG\$UNPASTE_VIRTUAL_DISPLAY Return Status—Problem Corrected

V5.5-2 In previous versions of VMS, if a virtual display was unpasted when the virtual cursor was not on the visible screen, the SMG\$UNPASTE_VIRTUAL_DISPLAY routine returned a SMG\$_INVROW status when it should have returned SMG\$_NORMAL. In VMS Version 5.5-2, this problem has been corrected.

3.8.38 TERMTABLE Print-Screen Definitions—Problem Corrected

V5.5-2 In previous versions of VMS, the print-screen definitions in the TERMTABLE files for SMG were incorrect for the VT200 and VT300 terminals. In VMS Version 5.5-2, this problem has been corrected.

3.9 SCSI Disk Class Driver Audio \$QIO Functions

V5.5-2 As part of its support for the RRD42 CDROM reader, VMS Version 5.5-2 provides a Small Computer System Interface (SCSI) disk class driver and \$QIO interface that have been extended to provide SCSI audio functions. This audio functionality is in addition to the standard Direct Access Device (Group 8) commands defined by the SCSI II standard (see the *VMS I/O User's Reference Manual: Part I*). For more information about these commands, see Appendix A.

3.10 System Services Notes

The release notes in this section pertain to VMS system services.

3.10.1 \$ADJWSL Argument—Problem Corrected

V5.5-2 VMS Version 5.0 added support for working sets larger than 65535 pages. The \$ADJWSL system service changed to support larger values in its **pagcnt** argument. Despite this, the **wsetlm** argument continued to return only word-length values.

In VMS Version 5.5-2, \$ADJWSL returns a longword value instead of a word value for the **wsetlm** argument. For most applications, this probably makes no difference. Applications written prior to this change may have reserved only two bytes for the **wsetlm** argument value. Depending on the use of the two bytes following that space, such applications may see unexpected results.

Applications that use the \$GETJPI system service to examine working set limits are unaffected by this change, as that service has always returned longword values.

3.10.2 \$CREMBX System Service—Maximum Allowable BUFQUO Value Changed

V5.5-2 Prior to VMS Version 5.5, the BUFQUO parameter to the \$CREMBX system service accepted a maximum value of 65355. In VMS Version 5.5-2, the maximum recommended value of BUFQUO is 60000. If the value is too high, the following error message appears:

```
-SYSTEM-F-BADPARAM, bad parameter value
```

This error message does not appear unless the value is set to 65324 or higher. However, it is recommended that programs use a value no higher than 60000. Values of 60000 or lower reduce the likelihood of programs failing because of future changes to VMS.

3.10.3 Return Codes from \$GETQUI and \$SNDJBC—Problem Corrected

V5.5-2

The following \$GETQUI and \$SNDJBC status return codes, which were incorrectly returned in VMS Version 5.5, have been corrected:

- JBC\$_NORMAL was returned instead of SS\$_NORMAL for a CANCEL_OPERATION function.
- JBC\$_JOBQUEDIS was returned in some cases instead of SS\$_NORMAL (JBC\$_JOBQUEDIS should appear only in the IOSB block).
- JBC\$_TOOMUCHINFO was returned instead of SS\$_MBTOOSML.

3.10.4 \$SNDJBC System Service—SJC\$_CHARACTERISTIC_NUMBER Problem Corrected

V5.5-2

In VMS Version 5.5, a problem existed with the SJC\$_CHARACTERISTIC_NUMBER item code of the \$SNDJBC system service.

If you specified the SJC\$_CHARACTERISTIC_NUMBER item code after any of the following item codes in the \$SNDJBC item list, the values of those item codes were lost:

- SJC\$_FILE_COPIES
- SJC\$_FILE_IDENTIFICATION
- SJC\$_FILE_SPECIFICATION

Jobs submitted under these conditions failed to execute or print because the file could not be found.

In VMS Version 5.5-2, this problem has been corrected.

3.10.5 \$START_TRANS—Status Return Codes Altered

V5.5-2

In previous versions of VMS, the \$START_TRANS system service returned the status SS\$_ABORT if either the local node did not have a transaction log or DECdtm services were disabled on the node.

In Version 5.5-2, \$START_TRANS no longer returns the status SS\$_ABORT. Instead, it returns:

- SS\$_NOLOG if the local node does not have a transaction log
- SS\$_TPDISABLED if DECdtm services are disabled on the local node

3.11 VAX BASIC Run-Time Library READ REGARDLESS Clause – Problem Corrected

V5.5-2

In previous versions of VMS, a problem occurred when you used a READ statement with a WAIT, followed by a READ with the REGARDLESS clause. The system ignored the REGARDLESS clause, and the READ caused the program to hang. In VMS Version 5.5-2, this problem has been corrected.

3.12 VAX C Run-Time Library Notes

The release notes in this section pertain to the VAX C Run-Time Library.

Programmer Release Notes

3.12 VAX C Run-Time Library Notes

3.12.1 Atof Function—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library function `atof()` failed to set `ERRNO` to the value `ERANGE` when the input string was a 2-digit number outside the range of representable values. In VMS Version 5.5-2, this problem has been corrected.

3.12.2 Ceil and Floor Functions—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library functions `ceil()` and `floor()` returned incorrect values for parameters close to 0. For example, `ceil(.998)` might return a value of 0 instead of the correct value of 1. In VMS Version 5.5-2, this problem has been corrected.

3.12.3 Longjmp Function—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library function `longjmp()` failed to convert a value parameter of 0 to 1. In VMS Version 5.5-2, this conversion problem has been corrected.

3.12.4 Math Functions—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library math functions `atan2`, `cabs`, `cosh`, `hypot`, `sinh`, and `tanh` did not intercept math errors or translate those errors to `ERRNO` values. In VMS Version 5.5-2, this problem has been corrected.

3.12.5 Qsort Function—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library function `qsort()` would often corrupt the data sorted after it switched from a longword sort to a byte sort, or vice versa. In VMS Version 5.5-2, the `qsort()` function no longer corrupts the data sorted.

3.12.6 Scanw Function—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library function `scanw()` failed to return the error value when it encountered EOF. It would return success, even when it should return error. In VMS Version 5.5-2, this problem has been corrected. The function `scanw()` correctly returns the error value.

3.12.7 SETVBUF Program—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library I/O routines had a problem in which the `SETVBUF` program corrupted output files. If a program called `SETVBUF` to increase the size of the output buffer, the output file that the program produced would receive false data appended to the end of the file when the file was closed. In VMS Version 5.5-2, the code that handles the flushing of data to a file on closure has been fixed to handle the larger buffer size correctly.

3.12.8 Socket Function Calls—Problems Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library socket functions interfered with exceptions, usually resulting in an inability to receive a `SIGALRM` signal to wake up a `socket()` call. In VMS Version 5.5-2, all exceptions are correctly delivered around socket function calls.

Also, in previous versions of VMS, several VAX C Run-Time Library socket functions showed incorrect return values. In VMS Version 5.5-2, this problem has been corrected.

Finally, in previous versions of VMS, the VAX C Run-Time Library I/O system sometimes did not deallocate memory correctly. Programs that repeatedly called the socket open and close functions eventually ran out of memory because the socket close function failed to deallocate part of the socket descriptor. In VMS Version 5.5-2, this problem has been corrected.

3.12.9 Strncat Function—Problem Corrected

V5.5-2 In previous versions of VMS, the VAX C Run-Time Library function `strncat()` incorrectly handled strings that were larger than 64K bytes in length. In VMS Version 5.5-2, this problem has been corrected.

3.13 VAX Text Processing Utility (VAXTPU)

The release notes in this section pertain to changes and corrections to the VAX Text Processing Utility (VAXTPU) and the EVE editing application within VAXTPU.

3.13.1 GET_INFO Built-In Procedure—New Parameter

V5.5-2 In VMS Version 5.5-2, the `GET_INFO` (*window_variable*) built-in procedure has the new parameter `screen_update`. This parameter allows applications to the update status of a window. Calling this built-in procedure returns either a 0, which implies that updates are turned off, or a 1, which implies that the updates are turned on.

3.13.2 READ_LINE Built-In Procedure—Problem Corrected

V5.5-2 In previous versions of VMS, the prompt area was not refreshed after you executed the VAXTPU built-in procedure `READ_LINE`. This problem occurred in multiline prompt areas or prompt areas that overlapped multiline windows. In VMS Version 5.5-2, this problem has been corrected.

DECwindows prompt areas are still limited to single lines, whereas character cell windows may have multiline prompt areas.

3.13.3 SCANL and SPANL Built-In Procedures—Problem Corrected

V5.5-2 In previous versions of VMS, VAXTPU had a problem in which the partial pattern assignment of the `SCANL()` and `SPANL()` built-in procedures caused a bugcheck to occur. This bugcheck occurred when the character causing the built-in procedures to stop matching appeared at the beginning of a line. In VMS Version 5.5-2, this problem has been corrected.

3.13.4 SEARCH Built-In Procedure—Problem Corrected

V5.5-2 In previous versions of VMS, when the VAXTPU built-in procedure `SEARCH` chose between two alternatives that matched at the same location, the built-in procedure chose the wrong alternative. This error was caused by a problem in the alternation pattern operator. In VMS Version 5.5-2, this problem has been corrected.

3.13.5 SET (SCREEN_UPDATE) Built-In Procedure—New Optional Parameter

V5.5-2 In VMS Version 5.5-2, the VAXTPU built-in procedure `SET (SCREEN_UPDATE)` has been modified to control the updates of windows within VAXTPU. The command to activate this built-in procedure now reads as follows:

```
SET (SCREEN_UPDATE, {ON/OFF} [window])
```

Programmer Release Notes

3.13 VAX Text Processing Utility (VAXTPU)

This built-in procedure now takes an optional third parameter, [window]. If you specify the OFF keyword and a window to the SET (SCREEN_UPDATE) built-in procedure, the resulting window is a no-update or clear window. VAXTPU does not acknowledge clear windows. Applications that modify part of the screen through some external means can map clear windows to that portion of the screen to prevent VAXTPU from overwriting the external data.

When you map a clear window to a buffer, you freeze that portion of the screen spanned by the window. In general, applications create clear windows and then perform no other operations upon the window. However, you can perform all normal window operations without error.

If you use the [window] parameter with SET (SCREEN_UPDATE, ON), the screen effects listed in Table 3-1 occur.

Table 3-1 Possible Screen Effects of [window] Parameter

Built-In Procedure	Effect
SCROLL, UPDATE SET (STATUS_LINE)	These built-in procedures have no effect on a clear window.
UNMAP, DELETE	If other windows are mapped to that screen area, the new windows determine what happens during the next update. If no windows are mapped to the area, VAXTPU erases previously unaffected lines.
ADJUST_WINDOW	If you increase the size of a window, VAXTPU still keeps the lines of text within that window on the screen. VAXTPU no longer modifies these lines, and you must use some other utility if you want to delete them.
POSITION	The clear window becomes the current window. This is a detached state, in that the cursor cannot reflect the editing point. You are unable to determine the position of the cursor on the screen.

Setting SET (SCREEN_UPDATE) to OFF takes effect immediately. Setting SET (SCREEN_UPDATE) to ON causes the window to be completely redrawn the next time you update the window.

3.13.6 SET (SCREEN_UPDATE) Built-In Procedure—Restrictions

V5.5-2

In VMS Version 5.5-2, the VAXTPU built-in procedure SET (SCREEN_UPDATE) has a [window] parameter. Note, however, that EVE does not support the use of clear windows. EVE often destroys, resizes, or creates windows to get certain screen effects, and does not acknowledge any clear windows. If you use applications with clear windows, do not use EVE windowing routines.

Note also that using the REFRESH, SET (SCREEN_UPDATE,ON), SET (HEIGHT), and SET (WIDTH) built-in procedures may cause a screen refresh. This destroys the screen context of the clear windows. You cannot avoid this screen refresh for all terminals; when you use applications possessing clear windows, you should avoid these built-in procedures.

3.13.7 New Security Option Bit Added to VAXTPU Callable Interface

V5.5-2 When an image is installed with application privileges and runs the VAXTPU EVE editor (either VAXTPU or the callable interface), unauthorized users can obtain those image privileges or use them in such a way as to violate system security.

Privileged applications calling VAXTPU should lower their application privileges before calling VAXTPU. If lowering the application privileges is not possible or desirable, privileged users are required to secure their applications against unauthorized access.

To secure an application or its accompanying privileges, or both, you must write a section file that prevents other users from accessing certain built-in procedures, such as CREATE_PROCESS, CALL_USER, SPAWN, WRITE_FILE, and READ_FILE.

By default, VAXTPU opens section files using all logical name tables. A new option bit, TPU\$V_SEC_LNM_MODE, has been added to the VAXTPU callable interface to allow applications to alter this default behavior. Privileged users are encouraged to lower their privileges instead of using this option, but may use this option to further secure their section file.

If set, the option bit TPU\$V_SEC_LNM_MODE causes VAXTPU to use executive-level logical names if VAXTPU opens the section file from a privileged image. When VAXTPU is called from an unprivileged image, TPU\$V_SEC_LNM_MODE has no effect.

The use of TPU\$V_SEC_LNM_MODE (or TPU\$M_SEC_LNM_MODE) requires the use of the VAXTPU full callable interface. For more information, see the documentation for the TPU\$INITIALIZE routine in the *VMS Utility Routines Manual*.

3.13.8 WRITE_FILE Built-In Procedure Output—Problem Corrected

V5.5-2 In previous versions of VMS, when you used the VAXTPU built-in procedure WRITE_FILE with a range that did not start at the beginning of a record, the output contained incorrect characters. In VMS Version 5.5-2, this problem has been corrected.

3.14 VMS Debugger

The release notes in this section pertain to the VMS Debugger.

3.14.1 Debug with VMS Linker—Problem Corrected

V5.5-2 In previous versions of VMS, a problem existed with the VMS Linker, which produced symptoms with large FORTRAN programs. The VMS Linker sometimes incorrectly set and reported the current module at startup or after a breakpoint. You could manually set the correct module, but that became tedious and time-consuming during extended debugger sessions, or when you restarted frequently. In VMS Version 5.5-2, this problem has been corrected.

3.14.2 Debugging Multi-Image Programs—Problem Corrected

V5.5-2 In previous versions of VMS, with some multi-image programs the debugger received an internal error during the processing of a SET IMAGE command. In such cases, the image could not be debugged. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes

3.14 VMS Debugger

3.14.3 Debugging Programs with LCK\$M_DEQALL Modifier – Suggested Workaround

V5.5-1 When an application includes the LCK\$M_DEQALL modifier in a \$DEQ system service call, the modifier breaks the communication links between the portion of the debugger in the user process (the kernel) and in the main process. The result is that the user's process stays in hibernate (HIB) state.

To work around this problem, debug the application using the limited one-process mode rather than the default or multiprocess mode. To set up one-process mode, issue the following command:

```
$ DEFINE DBG$PROCESS NONE
```

3.15 XQP and File System

This section describes the corrections to erase-on-delete, file truncation, and locking.

3.15.1 Erase-on-Delete Data Blocks Not Erased—Problem Corrected

V5.5-1 If a file marked for erase-on-delete was moved by IO\$_MOVEFILE, then the old data blocks were not erased, even if the volume was marked for erase-on-delete. The old data blocks are now erased.

3.15.2 File Truncation—Problem Corrected

V5.5-1 Incorrect file truncation was caused by invalid lock value block data. Two lock routines used data from a lock value block without first checking to see whether the lock value block was valid. The result was an inappropriate truncation of the file in question. The data check has been corrected.

3.15.3 Lock-Conversion Condition Caused System Failure—Problem Corrected

V5.5-1 A rare lock-conversion condition in the XQP caused a system failure. If a lock conversion was granted between the time the request was queued and a dequeuing was requested, the returned status from the dequeuing request caused an error.

This problem has been corrected.

3.15.4 Null-Mode Lock Request Caused System Failure—Problem Corrected

V5.5-1 A null-mode lock request in the XQP caused a system failure. If a null-mode lock request was not immediately granted, an error status was returned to the requesting routine, which then issued an error that caused a system failure. Null-mode lock requests are now properly granted.

3.15.5 SS\$_DEADLOCK Error Status Returned on File Creation or Extension—Problem Corrected

V5.5-2 A synchronization error that occurred in the XQP and File System while extending the INDEXF.SYS file occasionally caused the error status SS\$_DEADLOCK to be returned to applications during file creation or extension. Applications that use RMS to create or extend files could also receive the error status RMS_F_DEADLOCK. In VMS Version 5.5-2, this problem has been corrected.

Programmer Release Notes 3.15 XQP and File System

However, if a VMS Version 5.5-2 system is running in a VAXcluster with systems running earlier releases of VMS, the probability of an application receiving this error status is increased. This is especially true in cases where applications are simultaneously creating or extending files on the same volume from both VMS Version 5.5-2 systems and systems running earlier versions of VMS. Digital recommends that you update all VAXcluster nodes to VMS Version 5.5-2 in order to minimize the possibility of applications receiving this error.

Documentation Release Notes

This chapter describes corrections to specific manuals in the VMS documentation set.

4.1 VMS Audit Analysis Utility Manual

V5.5-2 On pages AUD-20 through AUD-22 of the *VMS Audit Analysis Utility Manual*, the /SELECT qualifier lists three keywords for selecting data packets from an audit log file when, in fact, a log file does not contain such packets. The keywords are as follows:

```
DEVICE_NAME=device
INSTALL=FILE=filename
OBJECT=IDENTIFICATION=file-id
```

Current releases do not generate a device name packet. If the object is a file, the device name becomes part of the object name.

The install file name is contained in an object packet. Since an installed file name appears within an object name packet, you can select the installed file using the expression OBJECT=(NAME=filename).

The OBJECT=IDENTIFICATION criterion is not supported.

4.2 VMS Device Support Reference Manual

V5.5-2 Chapter 3 of the *VMS Device Support Reference Manual* documents the function and use of various VMS driver kernel routines. The routine EXE\$DEANONPAGED (page 3-20) lists the various input requirements. This routine requires input IRP\$B_TYPE from the I/O request packet before the call. You should add the following note as an implicit input with IRP\$B_TYPE:

Note

Note that the MSB of field IRP\$B_TYPE must be 0, unless it defines a shared memory structure. A value of 1 in the MSB of this field causes a crash when you use non-shared memory.

4.3 VMS Network Control Program Manual

V5.5-2 In previous versions of VMS, the *VMS Network Control Program Manual* and the NCP Help text stated an incorrect range for the EXECUTOR DELAY FACTOR parameter. The correct range for this parameter is 16 to 255.

Documentation Release Notes

4.3 VMS Network Control Program Manual

Also, in previous versions of VMS, the *VMS Network Control Program Manual* and the NCP Help text stated an incorrect range for the EXECUTOR MAXIMUM BROADCAST NONROUTERS parameter. The correct range for this parameter is 1 to 1023.

4.4 VMS System Services Reference Manual

V5.5-2 In VMS Version 5.5, the *VMS System Services Reference Manual* states that when you specify the QUI\$_AFTER_TIME item code (page SYS-332), the \$GETQUI system service returns, as a quadword absolute time value, the time at or after which the job can execute.

This is true only if you submitted the job prior to the time specified to execute. If the specified time at submission has already passed, the job executes immediately and \$GETQUI returns the system time at which you submitted the job.

4.5 VMS Version 5.5 Upgrade and Installation Manual

Table 4-1 corrects Table A-2 from the *VMS Version 5.5 Upgrade and Installation Manual*. The table in that manual was incomplete, and contained some incorrect values.

Table 4-1 License Unit Requirement Table (LURT)

System Marketing Model	License Types by Code					
	VMS				SIP	LP
	A	B	C	D	E	F
VAX 11/730	10	NA	NA	NA	230	50
VAX 11/750	12	NA	NA	NA	230	100
VAX 11/780, 785	13	NA	NA	NA	230	100
VAX 6000-210, 6000-310	58	NA	NA	NA	230	300
VAXft 110	60	NA	NA	NA	230	100
VAXft 310, 410, 610	58	NA	NA	NA	230	300
VAX 6000-220, 6000-320, 6000-410	69	NA	NA	NA	230	600
VAX 6000-230, 6000-330, 6000-510	81	NA	NA	NA	400	900
VAX 6000-240, 6000-340, 6000-350, 6000-420, 7000-610, 10000-610	93	NA	NA	NA	400	1200
VAX 6000-540, 6000-640	170	NA	NA	NA	600	2400
VAX 6000-550, 6000-650	195	NA	NA	NA	600	2400

Key to License Type Codes and Values

- A—VMS Capacity or VMS Unlimited or Base
- B—VMS Server
- C—VMS Concurrent user
- D—VMS Workstations
- E—System integrated products
- F—Layered products
- NA—Not applicable

(continued on next page)

Documentation Release Notes
4.5 VMS Version 5.5 Upgrade and Installation Manual

Table 4-1 (Cont.) License Unit Requirement Table (LURT)

System Marketing Model	License Types by Code					
	VMS				SIP	LP
	A	B	C	D	E	F
VAX 6000-560, 6000-660	220	NA	NA	NA	600	2400
VAX 6000-610	81	NA	NA	NA	400	1200
VAX 8200, 8250	20	NA	NA	NA	230	100
VAX 8300, 8350	25	NA	NA	NA	230	200
VAX 8500	63	NA	NA	NA	230	400
VAX 8530	65	NA	NA	NA	230	400
VAX 8550, 8700, 8810	72	NA	NA	NA	400	600
VAX 8600, 8650	28	NA	NA	NA	230	400
VAX 8800, 8820	93	NA	NA	NA	400	1200
VAX 8830, 6000-360, 6000-430, 6000-520, 6000-620	119	NA	NA	NA	600	1800
VAX 7000-620, 10000-620	145	NA	NA	NA	600	1800
VAX 6000-440, 6000-450, 6000- 460, 6000-530	143	NA	NA	NA	600	2400
VAX 7000-630, 10000-630,	167	NA	NA	NA	600	2400
VAX 7000-640, 10000-640,	197	NA	NA	NA	600	2400
VAX 6000-630, 9000-210, 9000- 410	143	NA	NA	NA	600	2400
9000-420	241	NA	NA	NA	800	4800
VAX 9000-430	330	NA	NA	NA	800	4800
VAX 9000-440	386	NA	NA	NA	800	4800
MicroVAX II	18	NA	100	NA	230	50
MicroVAX 2000, 3100-10e, 3100- 20e, 3100-30, 3100-40, 3100-80,	18	NA	100	NA	230	20
MicroVAX 3100-90	60	NA	100	NA	230	20
MicroVAX 3500, 3600, 3800, 3900, 4000-200	60	NA	100	NA	230	200
MicroVAX 3300, 3400, 4000-100	60	NA	100	NA	230	100
VAX 4000-300, 4000-400	60	NA	100	NA	230	300
VAX 4000-500, 4000-600	60	NA	100	NA	400	900
VAXstation II, II/GPX	NA	NA	NA	100	50	10
VAXstation 2000, 2000/GPX	NA	NA	NA	100	50	10

Key to License Type Codes and Values

- A—VMS Capacity or VMS Unlimited or Base
- B—VMS Server
- C—VMS Concurrent user
- D—VMS Workstations
- E—System integrated products
- F—Layered products
- NA—Not applicable

(continued on next page)

Documentation Release Notes
4.5 VMS Version 5.5 Upgrade and Installation Manual

Table 4-1 (Cont.) License Unit Requirement Table (LURT)

System Marketing Model	License Types by Code					
	VMS				SIP	LP
	A	B	C	D	E	F
VAXstation 3100, 3200, 3500, 3520, 3540, 4000-60, 4000 VLC, 4000-90	NA	NA	NA	100	50	10
VAXserver 2000	NA	52	NA	NA	50	10
VAXserver 3300, 3400, 3500, 3600, 3900, 4000-200, 4000-300	NA	100	NA	NA	50	10
VAXserver 6000-210, 6000-310	NA	1443	NA	NA	230	200
VAXserver 6000-220, 6000-320, 6000-410, 6000-420	NA	1737	NA	NA	230	400
VAXserver 6000-510	NA	600	NA	NA	230	400
VAXserver 6000-520	NA	890	NA	NA	230	400
VAXserver 9000-110, 9000-310	143	143	NA	NA	600	900
VAXserver 9000-320	241	241	NA	NA	800	1200
VAXserver 9000-330	330	330	NA	NA	800	1200
VAXserver 9000-340	386	386	NA	NA	800	1200

Key to License Type Codes and Values

- A—VMS Capacity or VMS Unlimited or Base
- B—VMS Server
- C—VMS Concurrent user
- D—VMS Workstations
- E—System integrated products
- F—Layered products
- NA—Not applicable

Audio Extensions to the SCSI Disk Class Driver

V5.5-2 This appendix describes SCSI disk class driver audio commands and the \$QIO interface by which the VMS operating system provides audio functionality to the SCSI disk. You should be familiar with the American National Standard for Information Systems—Small Computer System Interface-2 specification and with the VMS \$QIO interface, and have basic knowledge of the characteristics of CD-ROM devices.

A.1 SCSI Audio Commands

Table A-1 lists the SCSI audio commands supported by the SCSI disk class driver.

Table A-1 SCSI Disk Class Driver Audio Commands

Command	Audio Function Code ¹	Description
Play Audio MSF	AUDIO_PLAY_AUDIO_MSF (5)	Requests the CD-ROM to begin an audio playback operation. The two required command arguments specify absolute starting and ending addresses of the playback in terms of minutes, seconds, and frame (MSF).
Play Audio Track	AUDIO_PLAY_AUDIO_TRACK (6)	Requests the CD-ROM to begin an audio playback operation. The two required command arguments specify the starting and ending tracks of the playback in terms of track number and index.
Play Audio	AUDIO_PLAY_AUDIO (4)	Requests the CD-ROM to begin an audio playback operation. The two required command arguments specify the starting logical block address (LBA) and the transfer count, in blocks, of the playback.
Pause	AUDIO_PAUSE (0)	Requests the CD-ROM to suspend any active audio operations. In response, the CD-ROM enters the hold-track state, muting the audio output after playing the current block.
Resume	AUDIO_RESUME (1)	Requests the CD-ROM to resume any active audio operations. In response, the CD-ROM exits the hold-track state and resumes playback at the block following the last block played.

¹Symbolic values for the function codes of SCSI audio commands are defined in SYS\$EXAMPLES:CDVERIFY.C. Numeric values appear within parentheses in this table column.

(continued on next page)

Audio Extensions to the SCSI Disk Class Driver

A.1 SCSI Audio Commands

Table A-1 (Cont.) SCSI Disk Class Driver Audio Commands

Command	Audio Function Code ¹	Description
Get Status	AUDIO_GET_STATUS (9)	Requests from the CD-ROM the status of the currently active playback operation, as well as the state of the current block. The Get Status command corresponds to the SCSI II Read Sub-channel command (READ SUBQ).
Set Volume	AUDIO_SET_VOLUME (11)	Requests the CD-ROM to adjust the output channel selection and volume settings for ports 0 through 3. The Set Volume command corresponds to the SCSI II Mode Select command for the CD-ROM Audio Control Parameters page.
Get Volume	AUDIO_GET_VOLUME (12)	Requests from the CD-ROM the output channel selection and volume settings for ports 0 through 3. The Get Volume command corresponds to the SCSI II Mode Sense command for the CD-ROM Audio Control Parameters page.
Prevent Removal	AUDIO_PREVENT_REMOVAL (2)	Prevents the removal of the CD caddy from the CD-ROM drive.
Allow Removal	AUDIO_ALLOW_REMOVAL (3)	Allows the removal of the CD caddy from the CD-ROM drive.
Get TOC	AUDIO_GET_TOC (10)	Requests from the CD-ROM a list of each track on the disk, including information about the audio or data contents of each track. Applications that require a detailed knowledge of the organization of a CD-ROM can use this function to obtain that information. The Get TOC command corresponds to the SCSI II Read TOC command.

¹Symbolic values for the function codes of SCSI audio commands are defined in SYS\$EXAMPLES:CDVERIFY.C. Numeric values appear within parentheses in this table column.

A.2 \$QIO Interface to Audio Functionality of the SCSI Disk Class Driver

To employ the audio functions of the RRD42 CD-ROM reader, the application program issues a call to the \$QIO system service using the following format:

```
status=SYS$QIO ([efn] ,[chan] ,func [,iosb] [,astadr] [,astprm]
[,p1] [,p2] [,p3] [,p4] [,p5] [,p6])
```

Arguments

[efn]
[chan]
[iosb]
[astadr]
[astprm]

These arguments apply to the \$QIO system service completion, not to device interrupt actions. For an explanation of these arguments, see the description of the \$QIO system service in the *VMS System Services Reference Manual*.

func

The IO\$_AUDIO function code allows the SCSI disk class driver to process SCSI audio commands.

Audio Extensions to the SCSI Disk Class Driver

A.2 \$QIO Interface to Audio Functionality of the SCSI Disk Class Driver

p1

Address of an Audio Control Block (AUCB). The \$QIO system service passes a SCSI audio command and command-specific control information to the SCSI disk class driver in the AUCB structure (see Section A.3).

p2

Size of the AUCB.

A.3 Defining an Audio Control Block (AUCB)

An application program that issues a call to the \$QIO system service that specifies the IO\$_AUDIO function code in the **func** argument must supply the address of an AUCB structure in the **p1** argument and its size in the **p2** argument.

An AUCB defines a specific SCSI audio command and provides the SCSI disk class driver with command-specific arguments and control information. Table A-2 defines the fields that appear in an AUCB; these fields are pictured in Figure A-1. See SYS\$EXAMPLES:CDROM_AUDIO.C for a code example that illustrates how an AUCB is defined in the C programming language.

Figure A-1 Audio Control Block (AUCB)

AUCB Version Number	Audio Function Code	0
Argument 1		4
Argument 2		8
Argument 3		12
Reserved		16
Destination Buffer Address		20
Destination Buffer Count		24
Destination Buffer Transfer Count		28
Operating System Command Status		32
SCSI Command Status (optional)		36
Sense Data Buffer Address (optional)		40
Sense Data Buffer Count (optional)		44
Sense Data Buffer Transfer Count (optional)		48
Reserved		52

ZK-4625A

Audio Extensions to the SCSI Disk Class Driver

A.3 Defining an Audio Control Block (AUCB)

Table A-2 Contents of Audio Control Block

Field	Use																
Audio Function Code	Numeric or symbolic code representing the audio function desired by the application program. (See Table A-1.) The application program must provide a valid audio function code for each operation.																
AUCB Version Number	Version of the AUCB and SCSI disk class driver audio interface. For the current version of the interface (for instance, that distributed with VMS Version 5.5-1) the value of this field should be 1. This field must <i>never</i> contain a zero.																
Argument 1	This field contains the first argument of the function specified in the Audio Function Code field. The following values are the valid values for each audio function.																
	<table border="1"> <thead> <tr> <th>Audio Function Code¹</th> <th>Field Contents</th> </tr> </thead> <tbody> <tr> <td>AUDIO_PLAY_AUDIO_MSF (5)</td> <td>Starting Frames (Sec<<8) (Min<<16)</td> </tr> <tr> <td>AUDIO_PLAY_AUDIO_TRACK (6)</td> <td>Starting (Track <<8) Index</td> </tr> <tr> <td>AUDIO_PLAY_AUDIO (4)</td> <td>Starting logical block address.</td> </tr> <tr> <td>AUDIO_GET_STATUS (9)</td> <td>0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.</td> </tr> <tr> <td>AUDIO_SET_VOLUME (11)</td> <td>Longword representing the values to be used to determine the new output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.</td> </tr> <tr> <td>AUDIO_GET_VOLUME (12)</td> <td>Longword to receive the current values determining output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.</td> </tr> <tr> <td>AUDIO_GET_TOC (10)</td> <td>0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.</td> </tr> </tbody> </table>	Audio Function Code ¹	Field Contents	AUDIO_PLAY_AUDIO_MSF (5)	Starting Frames (Sec<<8) (Min<<16)	AUDIO_PLAY_AUDIO_TRACK (6)	Starting (Track <<8) Index	AUDIO_PLAY_AUDIO (4)	Starting logical block address.	AUDIO_GET_STATUS (9)	0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.	AUDIO_SET_VOLUME (11)	Longword representing the values to be used to determine the new output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.	AUDIO_GET_VOLUME (12)	Longword to receive the current values determining output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.	AUDIO_GET_TOC (10)	0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.
Audio Function Code ¹	Field Contents																
AUDIO_PLAY_AUDIO_MSF (5)	Starting Frames (Sec<<8) (Min<<16)																
AUDIO_PLAY_AUDIO_TRACK (6)	Starting (Track <<8) Index																
AUDIO_PLAY_AUDIO (4)	Starting logical block address.																
AUDIO_GET_STATUS (9)	0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.																
AUDIO_SET_VOLUME (11)	Longword representing the values to be used to determine the new output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.																
AUDIO_GET_VOLUME (12)	Longword to receive the current values determining output channel selection and volume settings for CD-ROM ports 0 through 3. See Figure A-2 for an illustration of the format of this longword. Note that many CD-ROM drives do not support ports 2 and 3.																
AUDIO_GET_TOC (10)	0 if LBA format, 1 if MSF format. See the SCSI II specification for information about these formats.																

¹For any function code not listed in this table, this field contains a zero.

(continued on next page)

Audio Extensions to the SCSI Disk Class Driver

A.3 Defining an Audio Control Block (AUCB)

Table A-2 (Cont.) Contents of Audio Control Block

Field	Use										
Argument 2	This field contains the second argument of the function specified in the Audio Function Code field. The following are the valid values for each audio function:										
	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Audio Function Code¹</th> <th style="text-align: left;">Field Contents</th> </tr> </thead> <tbody> <tr> <td>AUDIO_PLAY_ AUDIO_MSF (5)</td> <td>Starting Frames (Sec<<8) (Min<<16)</td> </tr> <tr> <td>AUDIO_PLAY_ AUDIO_TRACK (6)</td> <td>Ending (Track<<8) Index</td> </tr> <tr> <td>AUDIO_PLAY_AUDIO (4)</td> <td>Transfer count in number of contiguous blocks to be played</td> </tr> <tr> <td>AUDIO_GET_TOC (10)</td> <td>Starting track.</td> </tr> </tbody> </table>	Audio Function Code ¹	Field Contents	AUDIO_PLAY_ AUDIO_MSF (5)	Starting Frames (Sec<<8) (Min<<16)	AUDIO_PLAY_ AUDIO_TRACK (6)	Ending (Track<<8) Index	AUDIO_PLAY_AUDIO (4)	Transfer count in number of contiguous blocks to be played	AUDIO_GET_TOC (10)	Starting track.
Audio Function Code ¹	Field Contents										
AUDIO_PLAY_ AUDIO_MSF (5)	Starting Frames (Sec<<8) (Min<<16)										
AUDIO_PLAY_ AUDIO_TRACK (6)	Ending (Track<<8) Index										
AUDIO_PLAY_AUDIO (4)	Transfer count in number of contiguous blocks to be played										
AUDIO_GET_TOC (10)	Starting track.										
Reserved	Must be zero.										
Destination Buffer Address	Address of the application program's buffer from which the status from a GET_STATUS or GET_TOC function is returned.										
Destination Buffer Count	Size in bytes of the destination buffer specified in the Destination Buffer Address field. For the GET_STATUS function, this field must contain the value 48 to receive complete status information. For the GET_TOC function, this field must contain the value 804 to receive full status. The SCSI disk class driver truncates the status data, if the destination buffer size is smaller than the size of the data.										
Destination Buffer Transfer Count	<p>The SCSI disk class driver returns to this field the actual number of bytes transferred to the buffer specified in the Destination Buffer Address field.</p> <p>Before accessing data returned by the GET_TOC or GET_STATUS commands, an application program must check the contents of this field to determine precisely how many bytes were returned by the CD-ROM.</p> <p>The application program initializes this field to zero.</p>										
Operating System Command Status	<p>VMS completion status of the SCSI audio function. The \$QIO system service returns this completion status in the I/O status block specified in the iosb argument to the \$QIO system service call. See Table A-3 for a description of these status codes.</p> <p>The application program initializes this field to zero.</p>										
SCSI Command Status (optional)	<p>SCSI status of the current operation. The SCSI disk class driver returns the SCSI status byte for the SCSI audio command described by this AUCB in the low byte of the low order word of this field. It returns the sense key in the low byte of the high order word. See the SCSI II specification for information regarding SCSI status and SCSI sense keys.</p> <p>The application program initializes this field to zero.</p>										

¹For any function code not listed in this table, this field contains a zero.

(continued on next page)

Audio Extensions to the SCSI Disk Class Driver

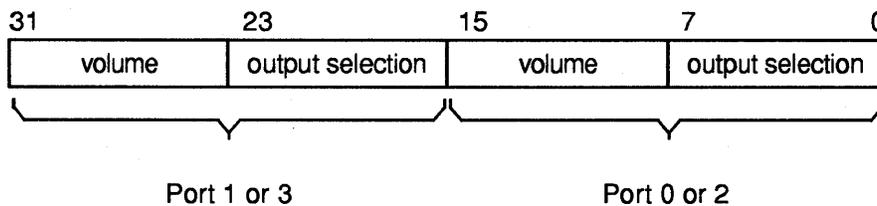
A.3 Defining an Audio Control Block (AUCB)

Table A-2 (Cont.) Contents of Audio Control Block

Field	Use
Sense Data Buffer Address (optional)	Address of buffer to which the SCSI disk class driver returns sense data when errors occur during audio function execution. When this field is specified, in the event of a check condition on an Audio command, the SCSI disk class driver automatically issues a Request Sense command to retrieve the Sense Key/Sense Data from the target. The target returns this data to the buffer specified in this field before the failing \$QIO audio function completes.
Sense Data Buffer Count (optional)	Size in bytes of the buffer specified in the Sense Data Buffer Address field. During request sense processing, the target device truncates the sense data to fit in this buffer.
Sense Data Buffer Transfer Count (optional)	Actual number of bytes of sense data returned to the application in the buffer specified in the Sense Data Buffer Address field. The application program initializes this field to zero.
Reserved	Must be zero.

The output channel selection and volume settings for CD-ROM ports as used by the SET_VOLUME function appear as shown in Figure A-2.

Figure A-2 Output Channel Selection and Volume Settings for CD-ROM Ports as Used by the SET_VOLUME Function



volume = 00 (muted) to FF (maximum)
 output selection <7:4> = 0
 output selection <3:0> = 0000 (output muted on this channel)
 0001 (connect audio channel 0 to this output port)
 0010 (connect audio channel 1 to this output port)
 0011 (connect audio channels 0 and 1 to this port)

ZK-4626A

A.4 Error Handling in Applications Using SCSI Audio Functions

As indicated in Table A-2, the AUCB provides for three levels of error status reporting:

- VMS condition values, returned in the Operating System Command Status field of the AUCB, as well as in the I/O status block specified by the **iosb** argument to the \$QIO system service call. (See Table A-3 for a description of these status codes.)

Audio Extensions to the SCSI Disk Class Driver

A.4 Error Handling in Applications Using SCSI Audio Functions

If this status is `SS$_NORMAL`, the function has completed without error. If the status is not `SS$_NORMAL`, the application program should check the SCSI Command Status field and the Sense Data buffer to fully determine the cause of the failure.

- SCSI command status, returned in the SCSI Command Status field of the AUCB. The SCSI disk class driver returns to this field SCSI status as well as the Sense Key in the event of a check condition SCSI status. The Sense Key can be used to determine the first level of error reporting supported by SCSI. See the SCSI II specification for further information.
- Sense data, returned in the buffer specified in the Sense Data Buffer Address field of the AUCB. Sense data bytes are assigned as defined in the SCSI II specification. Sophisticated programmers can use the data in this to obtain detailed information about the error-causing condition.

If the CD-ROM device is currently software enabled (that is, the volume has been mounted) and a unit attention is detected, then mount verification will be initiated by the driver. However, if the CD-ROM is not software enabled, the event will simply be returned to the application issuing the Audio \$QIO function.

Table A-3 VMS Status Codes Returned to the IOSB and AUCB by the SCSI Disk Class Driver

<code>SS\$_NORMAL</code>	AUCB command completed successfully.
<code>SS\$_ABORT</code>	Returned by the SCSI disk port driver. In general, you should retry commands that fail with this status.
<code>SS\$_BADPARM</code>	Driver detected an illegal value or missing value in the AUCB.
<code>SS\$_CTRLERR</code>	CD-ROM failed some part of its initialization sequence. When this status is returned, it is unlikely that the CD-ROM is usable.
<code>SS\$_DEVOFFLINE</code>	Device returned a not-ready sense key or failed the TEST UNIT READY/START sequence.
<code>SS\$_DRVERR</code>	CD-ROM failed to execute the command, either because the drive has failed or an illegal command was issued. Such a command could be a command that requested the drive to issue an audio command to a data track or attempted to perform a play operation on nonexistent tracks.
<code>SS\$_ILLIOFUNC</code>	Illegal I/O function was specified in the func argument of the \$QIO request.
<code>SS\$_IVADDR</code>	Specified block number is larger than <code>UCB\$_MAXBLOCK</code> .
<code>SS\$_MEDOFL</code>	Last command failed because the drive detected the removal and replacement of the CD carrier, or the unsuccessful completion of a request sense command after a check condition error. In general, you should not retry commands that fail with this status.

(continued on next page)

Audio Extensions to the SCSI Disk Class Driver

A.4 Error Handling in Applications Using SCSI Audio Functions

Table A-3 (Cont.) VMS Status Codes Returned to the IOSB and AUCB by the SCSI Disk Class Driver

SS\$_NOPRIV	Caller does not have sufficient privileges to execute this function. If the CD-ROM has not been mounted before an IO\$_READVBLK function is issued, this status may be returned.
SS\$_OPINCOMPL	Number of bytes requested is less than the number of bytes returned.
SS\$_PARITY	Nonrecoverable media error (does not apply to audio functions).
SS\$_RECOVERR	Recovered media error (does not apply to audio functions).
SS\$_VOLINV	CD-ROM has not been mounted.
SS\$_WRITLCK	Write operations not permitted on read-only devices.

A.5 Using CD-ROM to Store Both Data and Audio Information

In order to make effective use of mixed data and audio CDs, an application program requires detailed knowledge of the particular CD being played. The application program must know which tracks include data and which tracks include audio so that it can use commands appropriate to the different track types. Issuing audio command on a data track results in the command failing with a status of SS\$_DRVERR.

By default, the SCSI disk class driver transfers all requests issued to a CD-ROM in blocks of 512 bytes. This byte amount is referred to as the default block size. Before attempting to issue READ operations to the CD-ROM, you must ensure that the CD-ROM has been mounted as foreign or as a Files-11 volume. The application program can then determine, by issuing a GET_TOC function, which tracks (and, therefore, which logical blocks) contain data and which contain audio information.

A.6 Programming Audio Applications

The following list contains information useful in avoiding problems in writing code using the VMS SCSI audio interfaces:

- If the type of file system on the CD-ROM is not known, the user should mount the CD-ROM as foreign and issue a \$QIO request with the logical block I/O read function (IO\$_READLBLK) to read individual data blocks. The default block size for all CD-ROMs is 512 bytes.
- When using the GET_TOC command to obtain CD-ROM address information in LBA format, be advised that the byte ordering of the returned data is in big endian form. Because VAX byte ordering is little endian, you must switch the LBA data bytes to get a logical block address that is VAX valid. SYS\$EXAMPLES:CDVERIFYTEST.C contains an example of how to perform this exchange.
- Before attempting to issue a \$QIO request with the virtual block I/O read function (IO\$_READVBLK) to the CD-ROM, ensure that the CD-ROM has been mounted. Typically, non-Files-11 disks must be mounted foreign. If an IO\$_READVBLK \$QIO request is issued to an unmounted CD, the request fails with a status of SS\$_NOPRIV.

Audio Extensions to the SCSI Disk Class Driver

A.7 Application Program Example Using SCSI Audio Capabilities

A.7 Application Program Example Using SCSI Audio Capabilities

The file `SYS$EXAMPLES:CDROM_AUDIO.C` contains an example of an application program that performs the following tasks:

- Defines standard symbolic names for the audio function codes representing SCSI audio commands
- Defines representative AUCBs for each audio function code supported by the SCSI disk class driver
- Issues a series of QIO system service requests, each specifying the `IO$_AUDIO` function, that exercise the SCSI disk class driver to test its support for CD-ROM drives with audio capabilities
- Converts LBA data returned by a `GET_STATUS` command in big-endian byte-ordering form to VAX little-endian byte ordering form

New and Changed System Messages

This appendix alphabetically lists and describes new, updated, or previously undocumented system messages for the following facilities:

- BACKUP, Backup Utility
- EDT, EDT Editor
- JBC, Job Controller
- MAIL, Mail Utility
- QMAN, Queue Manager
- RMS, Record Management Services
- SYSTEM, System Services

For ease of reference, this appendix also includes any messages that were added in VMS Version 5.5-1.

CLOSEOUT, error closing 'filespec' as output

Facility: EDT, EDT Editor

Explanation: A VMS RMS error occurred while closing the specified output file. This message is accompanied by a VMS RMS message indicating the reason for the error.

For example, this message is accompanied by the RMS error ENT and the SYSTEM error NOPRIV if you attempted to exit from a file that does not have OWNER:DELETE privileges and a version limit exists for that file or for the directory in which the file is located.

The RMS error ENT occurs on an attempt to delete the .TMP file. The ENT error indicates that the version exceeds the version limit for the file or directory.

User Action: Follow the recovery procedure for the specified VMS RMS message.

For example, to correct the condition that produces the RMS error ENT, follow these steps:

1. Use the following command to recover your file:

EDIT/RECOVER filename

The journal (.JOU) file called by the EDIT/RECOVER command may not contain the last few edits that you made to the file. Update the file as necessary.

2. Press Ctrl/Z to obtain EDT's asterisk command prompt.

New and Changed System Messages

3. At the asterisk prompt, write your file to a new file with a different file name and then exit from the EDT editor. For example:

```
* WRITE NEWFILE.TXT  
* EXIT
```

The new file contains the edits that you made prior to the error.

To avoid this problem in the future, purge your files or use the SET FILE/VERSION command and qualifier to raise the version limit for the files or for the directory.

DME, dynamic memory exhausted

Facility: RMS, Record Management Services

Explanation: VMS RMS was unable to allocate additional process memory for the given request (usually an RMS \$OPEN or \$CONNECT system service call). Such requests can require a large number of buffers, large buffer sizes, or a combination of both.

User Action: Increase the amount of available process memory or decrease the size or number of requested buffers.

For process permanent files (such as DCL OPEN, DCL command procedures using "@filename," SYS\$INPUT, SYS\$OUTPUT, SYS\$ERROR, and batch log files), the size of available memory is governed by the SYSGEN parameter PIOPAGES. The number of buffers and their sizes is governed by the DCL command SET RMS. Only 63 process permanent files can be opened at once; any attempt to open more such files produces this message.

For image files, process memory is governed by the user's Authorization parameter PGFLQUOTA and the SYSGEN parameter VIRTUALPAGECNT. The size and number of buffers is controlled by the application and/or the DCL command SET RMS.

Image buffer space can also be controlled with the linker option of IOSEGMENT, which can determine the amount of fixed RMS memory allocated and the process region in which it is allocated. See the *VMS Linker Utility Manual* for further information.

DOTSPACK, error packing document files

Facility: MAIL, Mail Utility

Explanation: The packing process failed on an attempt to send a base DDIF file and one or more included files. The operation sent the text of the base DDIF file, but no included files were sent.

User Action: Check the base DDIF file and all included files for format errors. Correct any format errors and ensure that all included files are available; then retry the command.

DOTSUNPACK, error unpacking document files

Facility: MAIL, Mail Utility

Explanation: The unpacking process failed during an attempt to extract a DDIF file having one or more included files.

User Action: Check to see if you have enough disk space to unpack the file. If space is not the problem, you can ask the sender to try resending the files. Otherwise, you can extract the file and try to unpack it yourself using the appropriate CDA utilities.

New and Changed System Messages

INDEXTOSM, index file is too small on output device 'device-name'

Facility: BACKUP, Backup Utility

Explanation: The /NOINITIALIZE qualifier was specified on the output disk for an image backup, but the size of INDEXF.SYS on the output disk cannot accommodate the number of files on the input disk.

User Action: Initialize the output disk and increase the number of headers on the disk by using the /HEADERS and /MAXIMUM_FILES qualifiers on the INITIALIZE command. The output disk must contain as many or more headers as there are files on the input disk.

INVSMBMSG, invalid data in message from symbiont on queue 'queue-name' is being ignored

Facility: QMAN, Queue Manager

Explanation: A breach of symbiont or queue manager protocol has occurred. The queue manager has explicitly made adjustments to compensate for the breach. The symbiont will continue to process work.

This message is intended to alert symbiont writers to problems in symbiont code. In most cases, users other than symbiont writers need not be concerned when this message appears.

User Action: General users should report the occurrence of this message to the symbiont developer, or ask the system manager to do so.

Symbiont developers should review the symbiont code and the SMB and PSM utility routines in the *VMS Utility Routines Manual* to determine what activity is causing this message. Ensure that no improper state changes are being requested, such as stalling a starting queue or pausing a stopping queue. Check that all required messages are being provided and that no unnecessary messages are being issued.

NOUSERSPEC, no user specified; mail not sent

Facility: MAIL, Mail Utility

Explanation: You attempted to send a file using the DCL command MAIL. However, you failed to specify an addressee on the command line or at the To: prompt. The file was not sent.

User Action: Specify the address of the user to whom you wish to mail the file. You can enter the address on the command line or in response to the To: prompt.

PAGECRIT, page file nearly full; system trying to continue

Facility: SYSTEM, System Services

Explanation: The system is running out of page file space. This message is more critical than the PAGEFRAG message.

User Action: Create a new page file with more space. See the *Guide to Setting Up a VMS System*.

PAGEFRAG, page file filling up; please create more space

Facility: SYSTEM, System Services

Explanation: The system is running out of page file space.

User Action: Create a new page file with more space. See the *Guide to Setting Up a VMS System*.

New and Changed System Messages

QMANABORT, request failed because queue manager aborted

Facility: JBC, Job Controller

Explanation: A system service or DCL command has failed because the queue manager process terminated execution. If the queue manager encounters a fatal system error or internal logic error, it voluntarily aborts and is restarted. The command or system service call that returns this message may have caused the queue manager to abort.

User Action: Contact your system manager and report this error along with the command or system service call that returned the error. Digital recommends that you do not reissue the command or system service call that returned this error until the problem is resolved.

The system manager should submit a Software Performance Report (SPR) and include the following information:

- The dump file SYS\$SYSTEM:QMAN\$QUEUE_MANAGER.DMP from the node on which the queue manager was previously running
- A copy of any messages written to the console or the operator log file with the QUEUE_MANAGE or JOB_CONTROL user name

For information on how to submit an SPR, see the chapter on batch and print operations in the *Guide to Maintaining a VMS System*.

QUEDISABLED, disabled queue cannot be modified, nor can a job be submitted to it

Facility: JBC, Job Controller

Explanation: The queue you are trying to modify or to which you are trying to submit a job has been disabled. You cannot modify or submit jobs to a disabled queue.

This message can also indicate a corrupt queue record.

User Action: If you are trying to submit a job, submit it to another queue and notify the system manager.

If you are a system manager, search the operator log file for the following message:

```
QMAN-I-QUEDISCOR, queue 'name' has been disabled  
due to database corruption
```

This message indicates a corrupt queue record for the named queue. If you find this message, follow the instructions provided in the documentation for that message.

QUEDISCOR, queue 'name' has been disabled due to database corruption

Facility: QMAN, Queue Manager

Explanation: The queue manager has detected corruption in a queue record of the queue database. The related queue has been disabled in order to isolate the corruption. This message is written on the console and in the operator log file.

User Action: Notify the system manager.

If you are a system manager, submit a Software Performance Report (SPR) and include copies of the queue file and master file of the queue database. To copy the files while the queuing system is running, use the

New and Changed System Messages

CONVERT/SHARE command. For complete instructions, see the chapter on managing queues in the *Guide to Maintaining a VMS System*.

After you create copies of these files, delete the disabled queue and create a new queue to replace it. If references to the corrupt queue exist, be sure to remove those references before you delete the queue. For example, the corrupt queue might be named as the target of a generic queue.

VMS TURBOchannel Device Driver Support

V5.5-2

The TURBOchannel architecture is part of Digital's workstation open bus strategy that provides a high-performance I/O interconnect. TURBOchannel, supported by certain workstations, is defined as a synchronous, asymmetrical I/O channel for connecting optional devices. VMS now supports programming for such I/O connections by permitting the writing of TURBOchannel third-party device drivers. The support includes special TURBOchannel device driver I/O operating system routines. Two types of I/O operations supported for TURBOchannel device data transfers are:

- Direct memory access (DMA)
- Programmed I/O (PIO)

The VMS operating system routines specific to TURBOchannel map TURBOchannel address space for DMA and support the setup and delivery of device interrupts. The VMS architecture of the TURBOchannel interface is similar to that of other I/O subsystems, such as the UNIBUS and Q22-bus models, described in the *VMS Device Support Manual* and the *VMS Device Support Reference Manual*. In general, you structure a TURBOchannel device driver like a UNIBUS or Q22-bus driver.

This appendix describes the VMS features of the DMA interface, recommends coding strategies, and includes information on TURBOchannel driver naming and loading. Descriptions of VMS routines for the TURBOchannel interface are provided in Section C.7. If you are writing a TURBOchannel device driver, refer to both the *VMS Device Support Manual* and the *VMS Device Support Reference Manual* for basic driver design.

C.1 Hardware Environment

The TURBOchannel device support option is offered on the VAXstation 4000-60 system. A TURBOchannel (DWCTX) adapter that provides the hardware connection between the VAXstation processor pin-bus and the TURBOchannel device (see Figure C-1) resides in the option module slot. Only one TURBOchannel slot is supported. The base address of the option module slot is 3000 0000₁₆. The adapter features a scatter-gather map for certain DMA transfers and contains a 32-bit CSR (control and status register) located at address 3680 0000₁₆.

Two kinds of transactions are permitted on the TURBOchannel: a programmed I/O transaction and a DMA transaction. A programmed I/O read or I/O write transaction in the address range of 3000 0000 to 33FF FFFF₁₆ (a 64 MB map) is translated to a TURBOchannel device transaction (see Figure C-2). Setting bit 9 (Enable Map) of the CSR enables the scatter-gather map for mapped DMA transactions. For more information on DMA transactions, see Section C.2.

VMS TURBOchannel Device Driver Support

C.1 Hardware Environment

Figure C-1 VAXstation with a TURBOchannel Subsystem

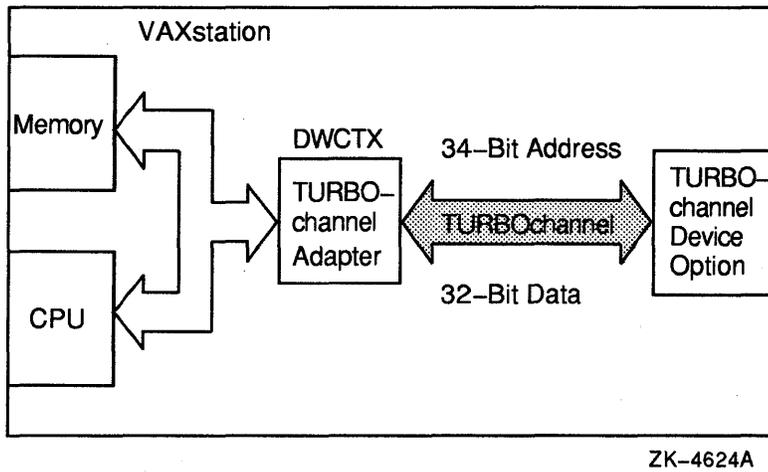
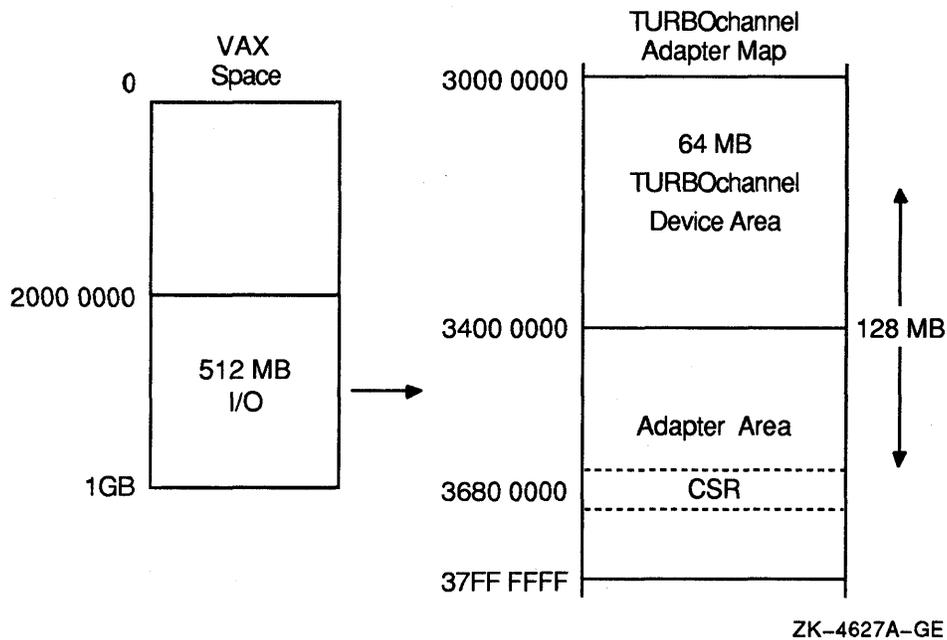


Figure C-2 TURBOchannel Map for the VAXstation CPU



C.1.1 Address Maps

Depending on the source of a transaction, the TURBOchannel adapter presents different address spaces to the components of the system. The workstation performs I/O transactions with devices in a physical address space of 512 MB starting at 2000 0000. However, TURBOchannel I/O occupies 128 MB of workstation I/O space starting at 3000 0000, as shown in Figure C-2. The first 64 MB of the map translate to the TURBOchannel device while the second 64 MB are reserved for adapter resources.

VMS TURBOchannel Device Driver Support C.1 Hardware Environment

When a TURBOchannel device is the source, the device performs a transaction into a 16 GB TURBOchannel physical space. In this direction, the adapter presents two maps to the TURBOchannel device. The two different maps are selected by enabling or disabling the adapter scatter-gather map as described in Section C.2.

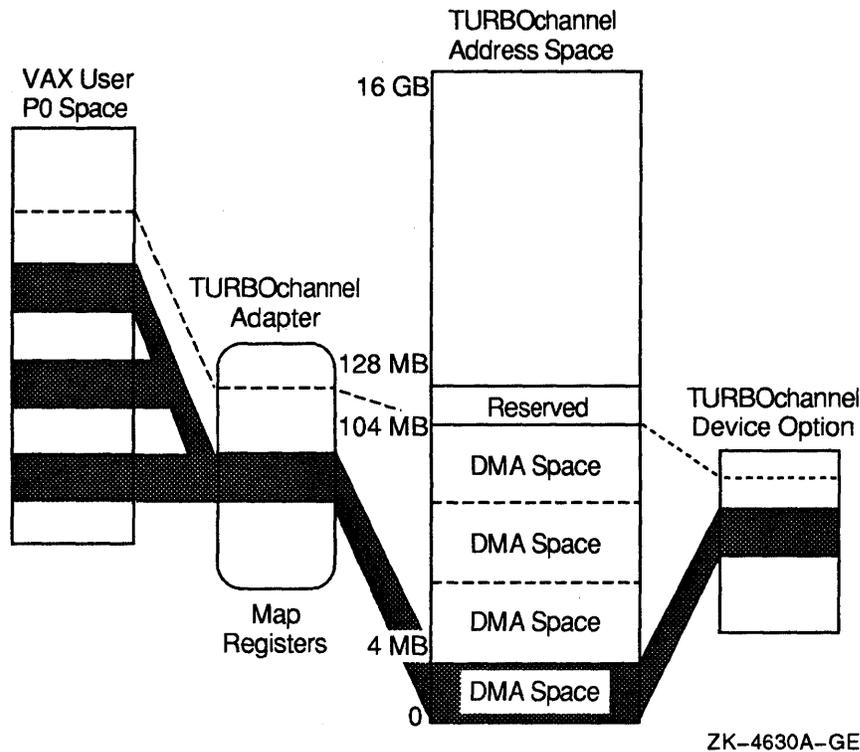
C.2 DMA Transactions

The direct memory access (DMA) I/O operation of a VAX host system permits devices and device drivers to exchange large amounts of data. DMA operations for TURBOchannel devices are similar to the Q22-bus DMA operations described in the *VMS Device Support Manual*. The TURBOchannel adapter sends 32-bit DMA data through the direct-DMA path between the VAX host and the TURBOchannel device. The direct data path (DDP) allows TURBOchannel transfers to randomly ordered physical addresses. The DDP maps each TURBOchannel I/O transfer to a pin-bus interconnect cycle.

As shown in Figure C-3, a TURBOchannel DMA device performs a DMA transfer toward a 16 GB physical address space of which there are two possible address maps:

- 4 MB mapped DMA space
- 104 MB unmapped DMA space

Figure C-3 TURBOchannel DMA to a VAX Host



VMS TURBOchannel Device Driver Support

C.2 DMA Transactions

C.2.1 4 MB Mapped DMA Space

When the scatter-gather map is enabled (see Section C.1), the adapter provides 4 MB of address space in which a TURBOchannel device can perform DMA transfers of contiguous pages that are mapped to discontinuous pages in VAX space (as shown in Figure C-3). Therefore, each page in the 4 MB space may be individually mapped to any page in the workstation's physical memory.

Note that the TURBOchannel adapter contains 8K map registers, each of which maps only 512 bytes (one VAX page each). Therefore, the lower 4 MB of TURBOchannel space is mapped to VMS address space if TURBOchannel DMA to VAX memory is performed. Note that images of the first 4 MB are reflected in the upper 4 MB regions.

C.2.2 104 MB Unmapped DMA Space

The 104 MB unmapped DMA space option allows a TURBOchannel device to perform DMA transfers with direct physical access to the workstation memory space. When the scatter-gather map is disabled (see Section C.1), the workstation's memory appears in the lower 104 MB of the TURBOchannel address space (as shown in Figure C-3). The memory space between this 104 MB and 128 MB boundary is reserved and should not be accessed by the TURBOchannel device. Note that images of the lower 128 MB are reflected in the upper 128 MB regions (because only the lower address bits are decoded).

C.2.3 Using TURBOchannel DMA Routines

There are three operating system routines provided for TURBOchannel DMA operations:

- IOC\$ALOTCMAP_DMA or IOC\$ALOTCMAP_DMAN
- IOC\$LOADTCMAP_DMA or IOC\$LOADTCMAP_DMAN
- IOC\$RELTCMAP_DMA or IOC\$RELTCMAP_DMAN

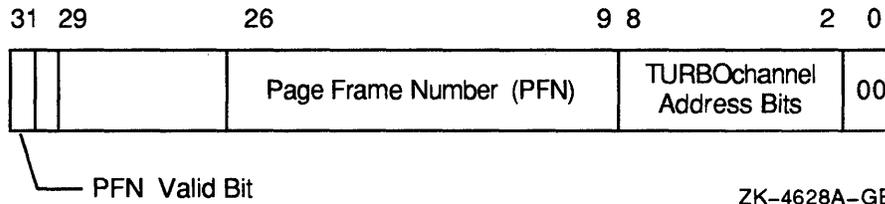
A driver that supports a device's DMA transfers to and from VAX memory must first allocate a set of map registers using an IOC\$ALOTCMAP routine. When DMA map registers are loaded, the IOC\$LOADTCMAP routines insert a VAX space PFN in each allocated map register. Note that one VAX page (512 bytes) of TURBOchannel device space is mapped into the VMS address space with each map register. As shown in Figure C-4, a field in each map register identifies the VAX page-frame number corresponding to the TURBOchannel space address that the map register represents.

Once the space is mapped, TURBOchannel devices are free to access this VMS memory with DMA read and write cycles. Once the DMA completes, the map registers are released (IOC\$RELTCMAP_DMA) to free the registers. Note that the _DMAN versions of the DMA routines are expedient for DMA operations with more than one map register and are passed information by way of the general purpose registers (instead of data structures). For more information on the DMA routines, see Section C.7.

VMS TURBOchannel Device Driver Support

C.3 Handling Interrupts

Figure C-4 TURBOchannel Map Register



C.3 Handling Interrupts

VAX peripheral devices request interrupts at IPLs 20 to 23 because device interrupts need to preempt most user and VMS software functions. Devices connected to the TURBOchannel subsystem should use IPL 20 as their device IPL to the workstation. In the reinitialization section established by the DPT_STORE macro, the driver prologue table holds the address of the interrupt service routine.

For further information on the interrupt service routine, refer to the *VMS Device Support Reference Manual*. The workstation uses non-direct-vector interrupt dispatching, which is described in the *VMS Device Support Manual*.

C.4 Coding a TURBOchannel Device Driver

You should write the device driver in one or more source files using the coding requirements described in the *VMS Device Support Manual*. Table C-1 lists standard driver routines for which you might need to provide entry points for the VMS operating system in your driver. The routines are described in more detail in the *VMS Device Support Reference Manual*.

VMS TURBOchannel Device Driver Support

C.4 Coding a TURBOchannel Device Driver

Table C-1 Driver Entry Point Routines

Routine	Description
Alternate Start I/O	Initiates activity on a device that can support multiple concurrent I/O operations and synchronizes access to its UCB.
Cancel I/O	Prevents further device-specific processing of the I/O request currently being processed on a device.
Controller Initialization	Prepares a controller for operation.
Driver Unloading	Prepares a driver for unloading or reloading.
FDT (\$QIO Handling)	Performs any device-dependent activities needed to prepare the I/O database to process an I/O request.
Interrupt Service	Processes interrupts generated by a device.
Register Dumping	Copies the contents of a device's registers to an error message buffer or a diagnostic buffer.
Start I/O	Activates a device to process a requested I/O function.
Timeout Handling	Takes whatever action is necessary when a device has not yet responded to a request for device activity and the time allowed for a response has expired.
Unit Delivery	For controllers that can control a variable number of device units, determines which specific devices are present and available for inclusion in the system's configuration.
Unit Initialization	Prepares a device for operation and, in the case of a device on a dedicated controller, initializes the controller.

The TURBOchannel support routines described in Section C.7 are supplied in a separate object library to which the driver is linked. When the driver is linked to include the TURBOchannel support routines (as described in the *VMS Device Support Manual*), the routines will reside in PSECT \$\$\$112_TC_SUPPORT of the resulting image. In the driver prologue table (DPTAB), set ADAPTER=TC to define the adapter type. For more information on device driver tables and other PSECTS, see the *VMS Device Support Manual*.

C.4.1 SYSGEN Autoconfigure

The System Generation Utility (SYSGEN) will autoconfigure a TURBOchannel device found in the TC slot. Third-party (unknown to Digital) TC devices found by SYSGEN will use the first and last letter in the option ROM name field of the device to build a driver name. During autoconfiguration, SYSGEN loads R4 with the physical address of the TC slot (3000 0000₁₆) during a call to the controller initialization or unit initialization routine for third-party TC devices.

C.5 Assembling and Linking a TURBOchannel Driver

Assemble the source files with (SYS\$LIBRARY:LIB.MLB), the system's macro library. For example:

```
$ MACRO QTDRIVER.MAR+SYS$LIBRARY:LIB.MLB/LIBRARY
```

Link the driver object file with the VMS global symbol table and the TURBOchannel routines object library. The global symbol table is located in SYS\$SYSTEM and is called SYS.STB, and the TURBOchannel routines are located in SYS\$LIBRARY:TC\$LIBRARY.OLB. If the driver consists of several source files, you must specify the file that contains the driver prologue table as

VMS TURBOchannel Device Driver Support

C.5 Assembling and Linking a TURBOchannel Driver

the first file in the list. The linker options file must contain a **BASE** statement specifying a zero base for the executable image.

The following is an example of how to link a TURBOchannel device driver with the TURBOchannel support routines:

```
$ CREATE QTDRIVER.OPT
BASE=0
Ctrl/Z
$ LINK /NOSYSSHR/NOTRACEBACK/NODEBUG/CONTIGUOUS QTDRIVER.OBJ,-
_$ SYS$LIBRARY:TC$LIBRARY/LIBRARY/SELECT,-
_$ QTDRIVER.OPT/OPTIONS,-
_$ SYS$SYSTEM:SYS.STB/SELECTIVE_SEARCH
```

The resulting image must consist of a single image section. The linker will report that the image has no transfer address; this report can safely be ignored. Note that if you do not use the TURBOchannel DMA routines, you do not need to include **TC\$LIBRARY** in the command line.

Once you have linked or relinked a driver, copy its image file to the **SYS\$LOADABLE_IMAGES** directory. By default, the **SYSGEN** commands **LOAD** and **CONNECT** search for a driver in the **SYS\$LOADABLE_IMAGES** directory.

C.6 Loading a TURBOchannel Device Driver

You can load a TURBOchannel device driver during the bootstrap program (for example, in **SYSTARTUP.COM**) or any time after the system is booted.

To load the driver into system virtual memory, run the System Generation Utility (**SYSGEN**) from the system manager's account or from an account with the **CMKRNL** privilege. Use commands in **SYSGEN** to load a TURBOchannel device driver and to create the device's I/O data structures. For more information on loading a driver with **SYSGEN**, refer to the *VMS Device Support Manual*.

Invoke **SYSGEN** by entering the following command:

```
$ RUN SYS$SYSTEM:SYSGEN
```

SYSGEN responds with the following prompt and waits for further input:

```
SYSGEN>
```

Use the **CONNECT** command (of **SYSGEN**) to load the driver and create the device's I/O database. You must specify the device name and the nexus number of both the TURBOchannel adapter and the TURBOchannel device.

You can obtain the adapter nexus number for the TURBOchannel adapter by issuing the **SHOW/ADAPTER** command, as shown in the following example:

```
SYSGEN> SHOW/ADAPTER
CPU Type: VAXstation 4000-60
Nexus (decimal)      Generic Name or Description
0000 0               KA46
0001 1               TURBOchannel adapter
```

As shown in the following example, the **SHOW/TURBOCHANNEL** command can be used to show the nexus number of your device on the TURBOchannel.

```
SYSGEN> SHOW/TURBOCHANNEL
TURBOCHANNEL: Device Name      Nexus Number      TC Slot
                   QMAT-AA          00000000          00000001
```

VMS TURBOchannel Device Driver Support

C.6 Loading a TURBOchannel Device Driver

The use of the `CONNECT` command to load a driver is shown in the following example:

```
SYSGEN> CONNECT QTA0/ADAPTER=1/TC_NEXUS=0/DRIVER=QTDRIVER
```

This command loads the driver `QTDRIVER` (if it is not already loaded) and creates the data structures needed to describe `QTA0`. It also calls the driver's controller and unit initialization routines for execution.

In this example, `QTA0` is the `SYSGEN` device designator and number (`QT` is your TURBOchannel driver mnemonic, `A0` is device #0). Note that Digital reserves driver names beginning with the letters *J* and *Q* for its customers. This ensures there are no conflicts with any other driver names on your system. Since `SYSGEN` examines the name area of your device module ROM to find names of active devices on a bus, you should prefix your device name in ROM with the letters *J* or *Q*. To determine a driver name, `SYSGEN` extracts the first and last letter in the name area of your ROM. For this example, `SYSGEN` extracts `QT` from the name `QMAT-AA` found in this device ROM.

Note that this example specifies a driver that has its device at Nexus 0 (`TC_NEXUS=0`) on the TURBOchannel and the TURBOchannel adapter at Nexus 1 (`ADAPTER=1`) on the CPU bus.

C.7 TURBOchannel Operating System Routines

This section describes the VMS operating system routines that are used by TURBOchannel device drivers supporting the connection between the VAX workstation bus and the TURBOchannel (adapter). The routines comprise the TURBOchannel DMA mapping interface.

IOC\$ALOTCMAP_DMA, IOC\$ALOTCMAP_DMAN

Allocates a set of TURBOchannel DMA map registers.

Module

[DRIVER]TCDMA_PTA

Input

Inputs for both routines follow:

Location	Contents
CRB\$L_INTD+VEC\$L_AD ADP	Address of ADP
ADP\$W_MRNREGARY ADP\$W_MRFREGARY ADP\$L_MRACTMDRS	Map register descriptor arrays
For IOC\$ALOTCMAP_DMA only	
R5	Address of UCB
UCB\$W_CRB	Address of CRB
UCB\$W_BCNT	Transfer byte count
UCB\$W_BOFF	Byte offset to start of transfer in first page
For IOC\$ALOTCMAP_DMAN only	
R1	Address of the map register descriptor (TC_MD)
R2	Address of ADP
R3	Number of map registers to be allocated

Output

Outputs for both routines follow:

Location	Contents
R0	SS\$_NORMAL or SS\$_INSFMAPREG
R2	Address of ADP
ADP\$W_MRNREGARY ADP\$W_MRFREGARY ADP\$L_MRACTMDRS	Updated
For IOC\$ALOTCMAP_DMA only	
R1	Destroyed
CRB\$L_INTD+VEC\$B_NUMREG	Number of map registers allocated
CRB\$L_INTD+VEC\$W_MAPREG	Starting map register number
For IOC\$ALOTCMAP_DMAN only	
R1	Address of the map register descriptor (TC_MD)

Operating System Routines IOC\$ALOTCMAP_DMA, IOC\$ALOTCMAP_DMAN

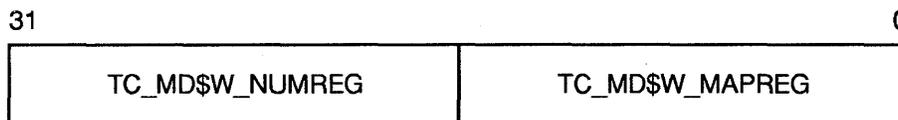
Synchronization

The caller of IOC\$ALOTCMAP_DMA or IOC\$ALOTCMAP_DMAN must be executing at fork IPL or above and must hold the corresponding fork lock (typically IOLOCK8) in a VMS multiprocessing environment. Either routine returns control to its caller at the caller's IPL. The caller retains any spin locks it held at the time of the call.

Description

IOC\$ALOTCMAP_DMA and IOC\$ALOTCMAP_DMAN allocate a contiguous set of TURBOchannel DMA map registers. IOC\$ALOTCMAP_DMA records the allocation in the ADP and CRB while IOC\$ALOTCMAP_DMAN records the same information in a map register descriptor. Figure C-5 shows the structure of the map register descriptor used by IOC\$ALOTCMAP_DMAN.

Figure C-5 TURBOchannel Map Register Descriptor (TC_MD)



ZK-4629A

TC_MD\$W_MAPREG contains the number of the first (starting) map register and TC_MD\$W_NUMREG contains the number of map registers allocated.

These routines differ in the way in which they determine the number of map registers they allocate:

- IOC\$ALOTCMAP_DMA calculates the number of needed map registers using the values contained in UCB\$W_BCNT and UCB\$W_BOFF.
- IOC\$ALOTCMAP_DMAN uses the value in R3 as the number of required registers.

If there are not enough contiguous map registers available, the routine returns an error status of SS\$_INSFMAPREG to its caller.

Because the map registers eventually must be released, the caller of IOC\$ALOTCMAP_DMAN must keep track of the map registers allocated. Care should be exercised in the consumption and management of map register resources.

When using the IOC\$ALOTCMAP_DMA routine, note that if there are not enough map registers available, your driver has the option to put a fork block onto the map register allocation wait queue in the ADP. When registers are released, the release routine checks for waiting fork threads. If any are waiting, the routine will attempt to complete the allocation at that time.

IOC\$LOADTCMAP_DMA, IOC\$LOADTCMAP_DMAN

Loads a set of TURBOchannel map registers for DMA.

Module

[DRIVER]TCDMA_PTA

Input

Inputs for both routines follow:

Location	Contents
For IOC\$LOADTCMAP_DMA only	
R5	Address of UCB
CRB\$L_INTD+VEC\$L_ADP	Address of ADP
UCB\$W_BCNT	Number of bytes in transfer
UCB\$W_BOFF	Byte offset to start of transfer in first page
UCB\$L_SVAPTE	System virtual address of PTE for first page of transfer
UCB\$L_CRB	Address of CRB
CRB\$L_INTD+VEC\$B_NUMREG	Number of map registers allocated
CRB\$L_INTD+VEC\$W_MAPREG	Number of first map register allocated
For IOC\$LOADTCMAP_DMAN only	
R1	Address of the map register descriptor (TC_MD shown in Figure C-5)
R2	Address of ADP
R3	System virtual address (SVAPTE) of first page to transfer
R4	Byte count of the transfer
R5	Byte offset to start of transfer in first page

Output

Outputs for both routines follow:

Location	Contents
R1, R2	Destroyed

Synchronization

A driver fork process calls IOC\$LOADTCMAP_DMA or IOC\$LOADTCMAP_DMAN at fork IPL, holding the corresponding fork lock (typically IOLOCK8) in a VMS multiprocessing environment. Either routine returns control to its caller at the caller's IPL. The caller retains any spin locks it held at the time of the call.

Operating System Routines

IOC\$LOADTCMAP_DMA, IOC\$LOADTCMAP_DMAN

Description

A driver fork process calls IOC\$LOADTCMAP_DMA or IOC\$LOADTCMAP_DMAN to load a previously allocated set of DMA map registers with page-frame numbers (PFNs). This enables a device to perform DMA transfer to or from the buffer indicated by the contents of UCB\$L_SVAPTE, UCB\$W_BCNT, and UCB\$W_BOFF (or the contents of R3, R4, and R5 when using IOC\$LOADTCMAP_DMAN).

Routine IOC\$LOADTCMAP_DMA or IOC\$LOADTCMAP_DMAN checks whether sufficient map registers were allocated. If there are insufficient map registers, the routine issues a UBMAPEXCED machine check. Otherwise, the routine loads the appropriate PFN into each map register.

IOC\$LOADTCMAP_DMA and IOC\$LOADTCMAP_DMAN load and set the mapping register valid for the number of mapping registers needed for the length of the DMA request.

IOC\$RELTCMAP_DMA, IOC\$RELTCMAP_DMAN

Releases a set of TURBOchannel DMA map registers.

Module

[DRIVER]TCDMA_PTA

Input

Inputs for both routines follow:

Location	Contents
ADP\$L_MRQFL	Head of queue of UCBs waiting for map registers
ADP\$W_MRNREGARY	Map register descriptor arrays
ADP\$W_MRFREGARY	
ADP\$L_MRACTMDRS	
For IOC\$RELTCMAP_DMA only	
R5	Address of UCB
UCB\$L_CRB	Address of CRB
CRB\$L_INTD+VEC\$L_ADP	Address of ADP
CRB\$L_INTD+VEC\$W_MAPREG	Starting map register number
CRB\$L_INTD+VEC\$B_NUMREG	Number of allocated map registers
For IOC\$RELTCMAP_DMAN only	
R1	Address of map register descriptor (TC_MD shown in Figure C-5)
R2	Address of ADP

Output

Outputs for both routines follow:

Location	Contents
R0	SS\$_NORMAL or SS\$_SSFAIL
R1	Destroyed
ADP\$W_MRNREGARY	Updated
ADP\$W_MRFREGARY	
ADP\$L_MRACTMDRS	

Synchronization

A driver fork process calls IOC\$RELTCMAP_DMA or IOC\$RELTCMAP_DMAN at fork IPL, holding the corresponding fork lock (IOLOCK8) in a VMS multiprocessing environment. Either routine returns control to its caller at the caller's IPL. The caller retains any spin locks it held at the time of the call.

Operating System Routines

IOC\$RELTCMAP_DMA, IOC\$RELTCMAP_DMAN

Description

A driver fork process calls IOC\$RELTCMAP_DMA or IOC\$RELTCMAP_DMAN to release a previously allocated set of the TURBOchannel DMA map registers.

IOC\$RELTCMAP_DMA obtains the location and number of the allocated map registers from CRB\$L_INTED+VEC\$W_MAPREG and CRB\$L_INTED+VEC\$B_NUMREG, respectively, while IOC\$RELTCMAP_DMAN obtains this same information from the map register descriptor (TC_MD).

After adjusting the map register descriptor arrays, IOC\$RELTCMAP routines examine the TURBOchannel DMA map-register wait queue. If the queue is empty, IOC\$RELTCMAP returns successfully to its caller. If the queue contains waiting fork processes, IOC\$RELTCMAP dequeues the first process and calls IOC\$ALOTCMAP_DMA to attempt to allocate the set of map registers it requires.

If IOC\$ALOTCMAP is called with sufficient map registers available, IOC\$RELTCMAP restores R3 through R5 to the process and reactivates it. When this fork process returns control to IOC\$RELTCMAP, IOC\$RELTCMAP attempts to allocate map registers to the next waiting fork process. IOC\$RELTCMAP continues to allocate map registers in this manner until the map-register wait queue is empty or it cannot satisfy the requirements of the process at the head of the queue. In the latter event, IOC\$RELTCMAP reinserts the fork process's UCB in the queue and returns successfully to its caller.

A

Access violation error, 3-1
Accounting Utility (ACCOUNTING), 2-11
ACMS, 3-1
AGEN\$FEEDBACK_REQ_TIME logical name, 2-2
ALL-IN-1
 and Batch and Print Queuing System, 2-7
Allocation classes
 in VAXcluster configurations, 2-1
Assisted copy operation
 See Copy assist
Assisted merge operation
 See Merge assist
Audio functions, 3-10
Autoconfiguration
 for TURBOchannel, C-6
AUTOGEN command procedure
 AGEN\$FEEDBACK_REQ_TIME logical name, 2-2
 new features, 2-1
 NUM_ETHERADAPT symbol, 2-2
 NUM_NODES symbol, 2-2
 problem when executed from SYSMAN, 2-1
 setting of TMSCP_LOAD parameter, 2-3
 specifying a minimum required age for feedback data, 2-2
 SYSMAN error, suggested workaround, 2-2

B

Backup Utility (BACKUP), 2-4
 alias directories, 2-5
 BACKUP/LIST operations, 2-4
 data compactions on tape drives, 2-4
 extending index files, 2-4
 message status codes, 2-5
 multivolume logical names, 2-5
 multivolume save operations, 2-5
 privileged users denied access to a tape, 2-4
 tape volume protection, 2-6
Batch and Print Queuing System
 and ALL-IN-1, 2-7
 chance of system shutting off, 2-7
 corruption detection for queues, 2-7

Batch and Print Queuing System (cont'd)
 DEFINE/CHARACTERISTIC command, 2-8
 DQS print symbiont, 2-8
 new system messages, B-1
 PRINT/USER command, 2-11
 queue manager CPU maximum limit, 2-10
 saving information in the queue database, 2-10
 \$SNDJBC problem fixed, 3-11
 stop_pending status, 2-10
 SUBMIT/USER command, 2-11
 SYMDEL message, 2-10
 SYNCHRONIZE command, 2-11
Batch job
 \$SNDJBC problem corrected, 3-11
 SYNCHRONIZE command, 2-11
Block transfer to a down node, 2-24
Bugcheck
 shadowing, 2-42

C

Cache, 2-31
Clear window, 3-13
Cluster, 2-13
CLUSTER_CONFIG.COM command procedure, 2-11
CNXMGRERR machine check, 2-24
Connection loss
 during shadowing state change, 2-42
Controlled terminals, 3-1
Controllers
 supporting performance assists, 2-33
 using performance assists, 2-33
Converting queues, 2-44
Copy assist
 controllers supporting, 2-34
 determining DCD connection limit, 2-38
 disabling, 2-37
 disabling on HSC controller, 2-39
 enabling, 2-37
 overview, 2-33
 performance, 2-34
 restriction, 2-40
 setting DCD connection limit, 2-39
 setting preferred path, 2-38

Corrupted queue database, 2-44
CPU maximum limit
for queues, 2-10
Ctrl/C and Ctrl/Y interrupts, 2-30

D

DCD (Disk-copy-data)
HSC connection limit, 2-38, 2-39
DCL
ON condition, 1-1
DCL commands
EDIT/TPU command /NOWORK qualifier, 1-1
READ command, 1-2
RECOVER, 2-29
SET (SCREEN_UPDATE) built-in procedure,
3-14
SET DEFAULT command, 1-2
SET MESSAGE command, 1-2
SET TERMINAL command, 1-3
SET TERMINAL/COMMSYNC, 1-3
SET TERMINAL/COMMSYNC command, 1-1
SET TERMINAL/MODEM command, 1-1
SHOW DEFAULT command, 1-2
SHOW DEVICE command, 1-3
TYPE command /PAGE qualifier, 1-4
DEBUG
SET IMAGE command, 3-15
DECnet-VAX, 2-20
event logger (EVL), 2-13
FDDI and Token Ring devices, 2-13
NCP module configurator, 2-19
NCP SHOW KNOWN NODES display, 2-20
NETACP quotas, 2-12
phase IV support for DSW-21 and DSW-41/42
devices, 2-20
DEC PHIGS
with DECwindows, 2-22
DECsound, 2-22
DECW\$MONITOR_DENSITY, 2-21
DECwindows
dashed lines incorrectly drawn, 2-20
monitor density, 2-21
server incompatibility, 2-22
SoftPC keyboard problem, 2-22
tiled operations, 2-23
use of SMP with DECsound and SODRIVER,
2-22
with DEC PHIGS, 2-22
X11 server, 2-20, 2-23
DEFINE/CHARACTERISTIC command, 2-8
Delta/XDelta Utility
booting new VAX computers, 2-23
DEQNA, 2-25
Detached recovery servers, 3-3
Device driver
TURBOchannel coding conventions, C-5

DIGITAL Storage Architecture
See DSA
Direct memory access (DMA)
TURBOchannel devices, C-3
TURBOchannel mapping, C-4
TURBOchannel map register, C-4
Disk class driver, 3-1
DSA, 3-1
Distributed lock manager, 2-24
LOCKMGRERR machine check, 2-25
lock range value corruption, 2-24
DMA map registers
for TURBOchannel, C-9, C-11, C-13
DMA routines
for TURBOchannel devices, C-4
DNS
patched images for Version 1.1, 2-20
Documentation corrections, 4-1
VMS Audit Analysis Utility Manual, 4-1
VMS Device Support Reference Manual, 4-1
VMS Network Control Program Manual, 4-1
VMS System Services Reference Manual, 4-2
DQS print symbiont, 2-8
DSA (DIGITAL Storage Architecture)
controllers, 3-1

E

Erase-on-delete, 3-16
Ethernet adapters
specifying number of, for AUTOGEN, 2-2
EVE editor key name definitions, 1-4
Event logger (EVL), 2-13
EXE\$DEANONPAGED routine
implicit input, 4-1
EXECUTOR NODE, 2-13

F

FDDI and Token Ring devices
specifying the number of Ethernet adapters for,
2-13
Feedback data
specifying a minimum required age, 2-2
FID (file identifier), 2-31
File identifier
See FID
Files
truncating, 3-16
File system and XQP
null-mode lock request, 3-16
File System and XQP
lock conversion condition, 3-16

H

- Hierarchical storage controller
 - See HSC
- HSC (hierarchical storage controller), 2-24
 - determining DCD connection limit, 2-38
 - disabling performance assists, 2-39
 - setting DCD connection limit, 2-39
 - setting preferred path, 2-38
 - supporting performance assists, 2-34

I

- Implicit target lists for generic queues, 2-9
- InfoServer
 - See PATHWORKS, 2-29
- InfoServer Client
 - startup behavior change, 2-25
- Interrupt
 - with TURBOchannel devices, C-5
- INVEXCEPTN error, 2-24
- IOC\$ALOTCMAP_DMAN routine, C-9
- IOC\$ALOTCMAP_DMA routine, C-9
- IOC\$LOADTCMAP_DMAN routine, C-11
- IOC\$LOADTCMAP_DMA routine, C-11
- IOC\$RELTCMAP_DMAN routine, C-13
- IOC\$RELTCMAP_DMA routine, C-13
- IPC\$SHARE image, 2-43

J

- JBCSYSQUE.DAT database, 2-44
- Jobs
 - saving information about, 2-10

K

- KDM70 controller
 - supporting performance assists, 2-34
- KLESI adapter, 2-31

L

- LAT
 - not supported on DEQNA, 2-25
- LATSYM
 - ACCVIO error, 3-1
- License Unit Requirement Table (LURT), 4-2
- Local tape devices
 - making available clusterwide TMSCP_LOAD parameter, 2-3
- Lock conversion condition in File System and XQP, 3-16
- Lost file header repair, 2-32

M

- Machine check
 - UBMAPEXCED, C-12
- Mailbox driver, 3-1
- Mail Utility (MAIL)
 - callable mail with SYSPRV, 3-1
 - DDIF and DTIF documents, 1-4
 - error message corrected, 1-4
- Mapped TURBOchannel DMA, C-4
- Map register
 - allocating for TC DMA, C-4
 - loading for TC DMA, C-4
- Mass storage control protocol
 - See MSCP
- Memory error, 2-32
- Memory requirements
 - for shadowing, 2-41
- Merge assist
 - See Minimerge operation
- Merge operation
 - unnecessary, 2-42
- Minimerge operation
 - controllers supporting, 2-34
 - disabling, 2-36
 - disabling on HSC controller, 2-39
 - enabling, 2-35
 - overview, 2-33
 - performance, 2-34
 - restriction, 2-40
 - write log entries, 2-36
- MOUNT/FOREIGN command, 2-27
- MOUNT/REBUILD command
 - avoids shadowing merge, 2-42
- Mount Utility, 2-25
 - bound volume set, 2-25
 - CPU halt caused, 2-26
 - hung other nodes, 2-26
 - logical names improperly defined, 2-27
 - multimember volume shadow set, 2-26
 - shadow set failure, 2-26, 2-27
 - shadow sets improperly allowed, 2-26
 - tape compaction problem, 2-27
 - tape density, 2-27
 - two-member volume shadow set, 2-26
- MSCP (mass storage control protocol) message, 2-24

N

- NCP (Network Control Program)
 - command prevents unauthorized connections, 2-17
 - module configurator, 2-19
 - SHOW KNOWN NODES display, 2-20
 - SHOW LINE display, 2-16

NDDRIVER, 2-19
NETACP, 2-13
 allocation of node counter blocks, 2-18
 endnode failover, 2-19
NETDRIVER, 2-19
New license types, 2-27
 Interactive User License, 2-28
 Operating System Base License, 2-27
Noncontiguous fields
 and lock range value corruption, 2-24
Non-modem port, 1-1
Null-mode lock request, 3-16
NUM_ETHERADAPT symbol, 2-2
NUM_NODES symbol, 2-2

O

OPCOM (Operator Communication Manager)
 process dump file, 2-28
 restarting, 2-28
Output job
 \$SNDJBC problem corrected, 3-11

P

PATH SPLIT POLICY, 2-19
PATHWORKS PC
 clients can interfere with InfoServer clients,
 2-29
Performance
 copy and merge assists, 2-33
 copy assist, 2-34
 disabling copy assists, 2-37
 disabling the minimerge, 2-36
 enabling copy assists, 2-37
 enabling the minimerge, 2-35
 minimerge, 2-34
 using write log entries, 2-36
Port command procedure, 3-2
Preferred path
 setting for HSC controllers, 2-38
 suggested workaround, 2-30
Print
 symbiont for DQS, 2-8
Print job
 \$SNDJBC problem fixed, 3-11
PRINT/USER command, 2-11
Programming
 EXE\$DEANONPAGED routine, 4-1
 TURBOchannel device driver, C-1
Pseudoterminal driver, 3-2
PTD\$READ routine, 3-2
PTD\$READW routine, 3-2
PTD\$WRITE routine, 3-2

Q

Queue
 conversion, 2-44
 database corrupted, 2-44
 detection of corruption, 2-7
 implicit target lists, 2-9
 SYNCHRONIZE command, 2-11
Queue database
 detection of queue corruption, 2-7
 saving job information, 2-10
Queue file
 detection of corruption, 2-7
Queue manager, 2-43
 clearing a stop pending condition, 2-10
 CPU maximum limit, 2-10
 detection of queue corruption, 2-7
 new system messages, B-1
 workaround for, 2-43

R

Race condition, 2-24
Recovery Unit Journaling
 See RMS Journaling
Remastering routine race condition, 2-24
RF35 controller
 supporting performance assists, 2-34
RF73 controller
 supporting performance assists, 2-34
RMS Journaling, 2-29, 3-2
 access to a remote indexed read-only file, 3-2
 deadlock among multiple detached recovery
 servers, 3-3
 detached recovery, 3-3
 invalid journals, 2-29
 with DCL command RECOVER, 2-29
RSPFATAL status error, 2-25

S

Scatter-gather map
 disabled, C-4
 enabled, C-4
Screen Management Run-Time Library (SMGRTL),
 3-3
 access violation, 3-4
 BLOCK_BORDER, 3-3
 incorrect output, 3-3
 input routine, 3-3
 line wrapping, 3-4
 SMG\$CHANGE_PBD_CHARACTERISTICS
 routine, 3-4
 SMG\$CHANGE_RENDITION routine, 3-4
 SMG\$CHANGE_VIEWPORT routine, 3-4
 SMG\$CHANGE_VIRTUAL_DISPLAY routine,
 3-4

Screen Management Run-Time Library (SMGRTL)
(cont'd)

SMG\$CREATE_MENU routine, 3-5
SMG\$DISABLE_UNSOLICITED_INPUT
routine, 3-5
SMG\$EXECUTE_COMMAND routine, 3-5
SMG\$INSERT_CHARS routine, 3-5
SMG\$INVALIDATE_DISPLAY routine, 3-5
SMG\$M_RETURN_IMMED routine, 3-5
SMG\$PASTE_VIRTUAL_DISPLAY routine,
3-5
SMG\$PRINT_PASTEBOARD routine, 3-6
SMG\$PUT_CHARS_HIGHWIDE routine, 3-6
SMG\$PUT_CHARS_MULTI routine, 3-6
SMG\$PUT_CHARS_WIDE routine, 3-6
SMG\$PUT_HELP_TEXT routine, 3-6
SMG\$PUT_LINE routine, 3-6
SMG\$PUT_LINE_HIGHWIDE routine, 3-7
SMG\$PUT_LINE_MULTI routine, 3-7
SMG\$PUT_STATUS_LINE routine, 3-7
SMG\$READ_COMPOSED_LINE routine, 3-7
SMG\$READ_STRING routine, 3-8
SMG\$READ_VERIFY routine, 3-8, 3-9
SMG\$REMOVE_LINE routine, 3-9
SMG\$SAVE_VIRTUAL_DISPLAY routine, 3-9
SMG\$SNAPSHOT routine, 3-9
SMG\$UNPASTE_VIRTUAL_DISPLAY routine,
3-10
TERMTABLE files, 3-10
viewport, 3-4
virtual display, 3-4
SCSI disk class driver
audio \$QIO functions, 3-10
Serving tape devices
TMSCP_LOAD parameter, 2-3
SHADDETINCOM error
shadowing bugcheck, 2-42
Shadow set, 2-27
failure of two-member, 2-26
improperly allowed by MOUNT, 2-26
MOUNT, 2-27
phase I race condition, 2-24
problem with bound volume set, 2-25
Shadow sets
supported, 2-42
SHOW ENTRY display, 2-11
SHOW QUEUE display, 2-11
Single system
enhanced shadowing performance, 2-33
SJC\$_CHARACTERISTIC_NUMBER item code
problem fixed, 3-11
SMISERVER process, 2-30
\$SNDJBC system service, 3-11
SODRIVER.EXE audio chip driver, 2-22
START/QUEUE command, 2-10
STOP/QUEUE/NEXT command, 2-10

Stop_pending condition
clearing with STOP/QUEUE/NEXT, 2-10
Striping
recommended version, 2-40
SUBMIT/USER command, 2-11
Symbiont for DQS, 2-8
SYMDEL message, 2-10
SYNCHRONIZE command, 2-11
System Generation Utility (SYSGEN)
LGI_BRK_TERM parameter, 2-30
System Management Utility (SYSMAN)
problem when executing AUTOGEN, 2-1
server hang, 2-30
System messages
new and changed, B-1
System parameters
PE1, 2-25
TMSCP_LOAD, 2-3
System services
\$ADJWSL, 3-10
\$CREMBX, 3-10
\$DEQ, 3-16
\$GETQUI, 3-11
\$SNDJBC, 3-11
\$START_TRANS, 3-11
System symbol definitions
high-level languages, 2-30

T

TA90E/91 tape drives, 2-27
Tape
compaction problem with MOUNT, 2-27
density improperly set with MOUNT/FOREIGN,
2-27
Tape devices
making available clusterwide TMSCP_LOAD
parameter, 2-3
Tape mass storage control protocol (TMSCP) server
software
TMSCP_LOAD parameter, 2-3
Terminals
controlled, 3-1
Tiled operations for DECwindows X11, 2-23
Timeout errors for DSA multiple controllers, 3-1
Tlan port, 1-3
TTDRIVER
modified to report error, 1-1
TUDRIVER
system failure during mount operation, 2-27
TURBOchannel device driver
adapter, C-1
assembling, C-6
autoconfiguration, C-6
coding, C-5
direct memory access, C-3
DMA routines, C-8
hardware environment, C-1

TURBOchannel device driver (cont'd)

- interrupt handling, C-5
- linking, C-6
- loading, C-7
- programming, C-1
- support, C-1

U

- UBMAPEXCED machine check, C-12
- UNIBUS adapter, 2-31
- Unmapped TURBOchannel DMA, C-4

V

VAX BASIC Run-Time Library

- READ REGARDLESS clause, 3-11

VAXcluster, 2-1

VAXcluster system

- data corruption on disks with new allocation classes, 2-11
- enhanced shadowing performance, 2-33
- recommended version, 2-40
- setting preferred path, 2-30
- shadowing restriction, 2-40
- specifying the number of nodes, for AUTOGEN, 2-2

VAX C Run-Time Library

- atof function, 3-12
- ceil function, 3-12
- floor function, 3-12
- longjmp function, 3-12
- math functions, 3-12
- qsort function, 3-12
- scanw function, 3-12
- SETVBUF program, 3-12
- socket functions, 3-12
- strncat function, 3-13

VAXstation

- 4000 Model keyboard problem, 1-4

VAXstations

- 4000 Model 60, 2-32
- 3100 Model 80, 2-32

VAX system

- 6000-600 and 6000-500 warm start, 2-31
- 6200/6300 performance, 2-31

VAXTPU

- EVE key name definitions, 1-4
- FILL built-in procedure, 1-5
- GET_INFO built-in procedure, 3-13
- READ_LINE built-in procedure, 3-13
- SCANL built-in procedure, 3-13
- SEARCH built-in procedure, 3-13
- SET (SCREEN_UPDATE) built-in procedure, 3-13, 3-14
- SPANL built-in procedure, 3-13
- tabs in overstrike mode, 1-5

VAXTPU (cont'd)

- TPU\$V_SEC_LNM_MODE option bit, 3-15
- work files, 1-5
- WRITE_FILE built-in procedure, 3-15
- VCB (volume control block), 2-24
- Verify Utility, 2-31
 - file headers, 2-31
 - lost file header repair, 2-32
- VMS Audit Analysis Utility, 2-1
- VMS Backup Utility
 - See Backup Utility (BACKUP)
- VMS Debugger, 3-15
- VMS Linker
 - symptoms in Debug, 3-15
- VMS Volume Shadowing, 2-33
 - and striping, 2-40
 - connection loss, 2-42
 - copy assist performance, 2-34
 - determining DCD connection limit, 2-38
 - disabling copy assists, 2-37
 - disabling performance assists, 2-39
 - disabling the minimerge, 2-36
 - enabling copy assists, 2-37
 - enabling the minimerge, 2-35
 - memory requirements, 2-41
 - minimerge performance, 2-34
 - number of supported shadow sets, 2-42
 - performance assists, 2-33
 - recommended VAXcluster version, 2-40
 - restrictions, 2-40
 - setting DCD connection limit, 2-39
 - supported controllers, 2-34
 - unnecessary merge operation, 2-42
 - write log entries, 2-36

Volume control block

See VCB

Volume Shadowing Phase II

- number of supported shadow sets increased, 2-42
- performance-assisted copy operation, 2-33
- performance-assisted merge operation, 2-33

Volume shadow set

See Shadow set

W

Warm start, 2-31

WHE

See Write log entries

Workaround

for SET PREFERRED PATH option, 2-30

Workaround for queue manager, 2-43

Workstation failure, 2-32

Write log entries

for minimerge operation, 2-36

X

XQP and File System

- erase-on-delete, 3-16
- in mixed version VAXclusters, 2-45
- lock conversion condition, 3-16
- null-mode lock request, 3-16
- SS\$_DEADLOCK error status, 3-16
- truncating files, 3-16

Y

- YEDRIVER, 1-3

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825) and press 2 for technical assistance.

Electronic Orders

If you wish to place an order through your account at the Electronic Store, dial 800-234-1998, using a modem set to -2400 or -9600 baud. You must be using a VT terminal or terminal emulator set at 8 bits, no parity. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825) and ask for an Electronic Store specialist.

Telephone and Direct Mail Orders

From	Call	Write
U.S.A.	DECdirect Phone: 800-DIGITAL (800-344-4825) FAX: (603) 884-5597	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	Phone: (809) 781-0505 FAX: (809) 749-8377	Digital Equipment Carribean, Inc. 3 Digital Plaza, 1st Street Suite 200 Metro Office Park San Juan, Puerto Rico 00920
Canada	Phone: 800-267-6215 FAX: (613) 592-1946	Digital Equipment of Canada Ltd. 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	_____	Local Digital subsidiary or approved distributor
Internal Orders ¹ (for software documentation)	DTN: 241-3023 (508) 874-3023	Software Supply Business (SSB) Digital Equipment Corporation 1 Digital Drive Westminster, MA 01473
Internal Orders (for hardware documentation)	DTN: 234-4325 (508) 351-4325 FAX: (508) 351-4467	Publishing & Circulation Services Digital Equipment Corporation NR02-2 444 Whitney Street Northboro, MA 01532

¹Call to request an Internal Software Order Form (EN-01740-07).

Reader's Comments

VMS Version 5.5-2
Release Notes
AV-PQZAB-TE

Your comments and suggestions help us improve the quality of our publications.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (product works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

I would like to see more/less _____

What I like best about this manual is _____

What I like least about this manual is _____

I found the following errors in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

For software manuals, please indicate which version of the software you are using: _____

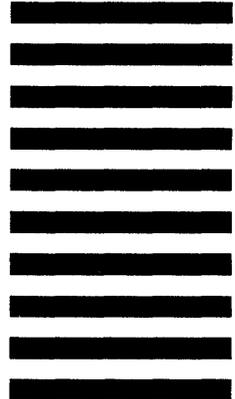
Name/Title _____ Dept. _____
Company _____ Date _____
Mailing Address _____
_____ Phone _____

----- Do Not Tear - Fold Here and Tape -----

digital™



No Postage
Necessary
if Mailed
in the
United States



BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
Information Design and Consulting
ZK01-3/J35
110 SPIT BROOK RD
NASHUA, NH 03062-9987



----- Do Not Tear - Fold Here -----